21世纪高等学校规划教材 | 计算机科学与技术





信息安全原理与应用技术

张健 任洪娥 黄英来 郭继峰 李三平 编著

清华大学出版社

21 世纪高等学校规划教材·计算机科学与技术

信息安全原理与应用技术

张　健　任洪娥
黄英来　郭继峰　李三平　编著

清华大学出版社
北　京

内 容 简 介

信息安全已经成为国家的重要战略，信息安全技术涵盖了信息与网络的方方面面。本书将从密码学与古典方法、分组密码体制、公钥密码体制、序列密码体制等算法以及操作系统安全、计算机病毒木马、入侵检测技术、无线网络安全等方面对信息安全原理与技术进行讲解。

本书是作者在多年教学和科研工作的基础上形成的，语言简练，通俗易懂，重点突出。该书可以作为高等学校计算机、通信工程、信息安全等专业的本科生和硕士生教材，也可以供从事相关领域的研究人员及工程技术人员参考。

图书在版编目(CIP)数据

信息安全原理与应用技术/张健，任洪娥等编著. —北京：清华大学出版社，2015
21世纪高等学校规划教材·计算机科学与技术
ISBN 978-7-302-41599-2

Ⅰ. ①信… Ⅱ. ①张… ②任… Ⅲ. ①信息安全－高等学校－教材 Ⅳ. ①TP309

中国版本图书馆CIP数据核字(2015)第225355号

责任编辑：郑寅堃 薛 阳
封面设计：傅瑞学
责任校对：白 蕾
责任印制：

出版发行：清华大学出版社
网 址：http://www.tup.com.cn，http://www.wqbook.com
地 址：北京清华大学学研大厦A座 **邮 编**：100084
社 总 机：010-62770175 **邮 购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课件下载：http://www.tup.com.cn，010-62795954
印 刷 者：
装 订 者：
经 销：全国新华书店
开 本：185mm×260mm **印 张**：13 **字 数**：315千字
版 次：2015年11月第1版 **印 次**：2015年11月第1次印刷
印 数：1～ 000
定 价： .00元

产品编号：063141-01

出版说明

随着我国改革开放的进一步深化，高等教育也得到了快速发展，各地高校紧密结合地方经济建设发展需要，科学运用市场调节机制，加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度，通过教育改革合理调整和配置了教育资源，优化了传统学科专业，积极为地方经济建设输送人才，为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是，高等教育质量还需要进一步提高以适应经济社会发展的需要，不少高校的专业设置和结构不尽合理，教师队伍整体素质亟待提高，人才培养模式、教学内容和方法需要进一步转变，学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007 年 1 月，教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》，计划实施“高等学校本科教学质量与教学改革工程”(简称“质量工程”)，通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容，进一步深化高等学校教学改革，提高人才培养的能力和水平，更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中，各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势，对其特色专业及特色课程(群)加以规划、整理和总结，更新教学内容、改革课程体系，建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上，经教育部相关教学指导委员会专家的指导和建议，清华大学出版社在多个领域精选各高校的特色课程，分别规划出版系列教材，以配合“质量工程”的实施，满足各高校教学质量和教学改革的需要。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作，提高教学质量的若干意见》精神，紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”，在有关专家、教授的倡议和有关部门的大力支持下，我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”)，旨在配合教育部制定精品课程教材的出版规划，讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师，其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求，“编委会”一致认为，精品课程的建设工作从开始就要坚持高标准、严要求，处于一个比较高的起点上。精品课程教材应该能够反映各高校教学改革与课程建设的需要，要有特色风格、有创新性(新体系、新内容、新手段、新思路，教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展，顺应并符合 21 世纪教学发展的规律，代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐)，经“编委会”认真评审，最后由清华大学出版

社审定出版。

目前，针对计算机类和电子信息类相关专业成立了两个“编委会”，即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括：

(1) 21世纪高等学校规划教材·计算机应用——高等学校各类专业，特别是非计算机专业的计算机应用类教材。

(2) 21世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 21世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。

(4) 21世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。

(5) 21世纪高等学校规划教材·信息管理与信息系统。

(6) 21世纪高等学校规划教材·财经管理与应用。

(7) 21世纪高等学校规划教材·电子商务。

(8) 21世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力，在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌，为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格，这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会
联系人：魏江江
E-mail：weijj@tup.tsinghua.edu.cn



前　言

随着通信和计算机技术的快速发展，以及经济全球化应用的推动，互联网表现出了极大的使用方便性和信息传递的快捷性，这使得人们对信息网络的依赖程度越来越高。人们在传递信息的同时，信息的安全性自然成为所关心的重要问题。

信息安全作为国家的重大战略，其意义不言而喻。信息安全也关乎每个人信息的安全。

作者根据多年教学经验和科研经验，在学习和总结国内外相关文献的基础上，完成了本书的撰写工作。

本书的特色是用通俗易懂的语言，对信息安全技术中的基本原理和技术进行准确阐述，并配合适当的例题进行深入研究，包括各种密码体制，以及信息安全在日常应用中的诸多应用安全技术。

全书共分为12章。第1章是信息安全概述；第2章至第8章介绍密码学相关技术；第9章介绍操作系统应用中的安全技术；第10章介绍计算机病毒和木马的相关概念和技术；第11章介绍入侵检测技术；第12章介绍无线网络安全技术。

本书的第1章和第2章由任洪娥编写，第3～5章由张健编写，第6章和第7章由黄英来编写，第8章和第9章由郭继峰编写，第10～12章由李三平编写，张健负责全书的统编。

为配合本课程的教学需要，本教材为教师配有习题参考答案，可发E-mail(ZhengYK@tup.tsinghua.edu.cn)联系索取。

作者要特别感谢参考文献中所列的各位作者，是他们的独到见解为本书提供了宝贵的资料及丰富的写作源泉。限于作者的水平和学识，书中难免存在疏漏和错误之处，诚望读者不吝赐教。

最后，谨向每一位关心和支持本书编写工作的各方面人士表示感谢！清华大学出版社为本书的出版做了大量的工作，在此表示衷心的谢意！

作　者

2015年10月

目 录

第1章 信息安全概述

随着计算机网络的不断发展,全球信息化已成为人类发展的大趋势。但由于计算机网络具有连接形式多样性、终端分布不均匀性和网络的开放性、互联性等特征,致使网络易受黑客、怪客、恶意软件和其他攻击,所以网络上信息的安全和保密是一个至关重要的问题。

1.1 信息安全与网络安全

信息安全是指信息网络的硬件、软件及其系统中的数据受到保护,不受偶然的或者恶意的原因而遭到破坏、更改、泄露,系统连续可靠正常地运行,信息服务不中断。

信息安全是一门涉及计算机科学、网络技术、通信技术、密码技术、信息安全技术、应用数学、数论、信息论等多种学科的综合性学科。

计算机网络最重要的方面是向用户提供信息服务及其所拥有的信息资源,网络安全从其本质上来讲是指网络上信息的安全,可将网络信息分为静态信息和动态信息,静态信息为存储于网络节点上的信息资源,将传播于网络节点间的信息,称为动态信息。国际标准化组织(ISO)将"计算机的安全"定义为"为数据处理系统建立和采取的技术和管理的安全保护,保护计算机硬件、软件数据不因偶然和恶意的原因而遭到破坏、更改和泄漏",此概念偏重于静态信息保护。此外"计算机安全"亦被定义为:"计算机的硬件、软件和数据受到保护,不因偶然和恶意的原因而遭到破坏,更改和泄露,系统连续正常运行。"该定义着重于动态信息的描述。网络安全本质上是信息安全的引申,网络安全是对网络信息保密性、完整性、可用性以及真实性的保护。其本质是在信息的安全期内保证信息在网络上流动时或者静态存放时不被非授权用户非法访问,但授权用户可以访问。

1.2 网络面临的安全威胁

网络必须有足够强的安全措施,否则网络将是个无用、甚至会危及国家安全的网络。无论是在局域网还是在广域网中,都存在着自然和人为等诸多因素的脆弱性和潜在威胁。因此,网络的安全措施应能全方位地针对各种不同的威胁和脆弱性,这样才能确保网络信息的保密性、完整性和可用性。

网络安全就是网络上的信息安全,涉及的领域很广。网络安全是指网络系统的硬件、软件及其系统中的数据受到保护,不因为偶然的或者恶意的原因而遭到破坏、更改、泄露,系统

连续可靠正常地运行，网络服务不中断，包括以下含义：

(1) 网络运行系统安全；

(2) 网络上系统信息的安全；

(3) 网络上信息传播的安全，即信息传播后的安全；

(4) 网络上信息内容的安全。

网络安全具有5个要素。

(1) 可用性：授权实体有权访问数据；

(2) 机密性：信息不暴露给未授权实体或进程；

(3) 完整性：保证数据不被未授权修改；

(4) 可控性：控制授权范围内的信息流及操作方式；

(5) 可审查性：对出现的安全问题提供依据与手段。

网络安全的内容有：

(1) 物理安全；

(2) 网络安全；

(3) 传输安全；

(4) 应用安全；

(5) 用户安全。

网络设计之初是为了方便信息的交流与开放，实现网络资源数据的共享，而对于保障信息安全方面的规划则非常有限。伴随计算机与通信技术的迅速发展，由于各种原因，网络面临着各式各样的安全威胁，诸如灾害(火灾、雷击、地震等)，网络结构的缺陷，一些恶意攻击(窃密、重放、篡改等)，以及软件漏洞等。这些最为主要的威胁导致网络固有的优越性、开放性和互联性，变成了信息安全性隐患的便利渠道。安全的目的是将计算机系统中的服务与资源的任何弱点降到最低限度，即将计算机系统的脆弱性降低到最低程度。网络安全问题与网络的脆弱性紧密相关，其脆弱性体现在以下几点。

1. 软件的脆弱性

随着软件规模的不断扩大，各种系统软件、应用软件也变得越来越复杂，只要有软件，就有可能存在漏洞，例如从 Windows 98 到 Windows XP、Windows 7 、Windows 8，各种版本的操作系统都有存在于操作系统脚本引擎中的安全漏洞，该漏洞能让黑客利用电子邮件或者恶意网站控制受害者的机器。此外除了 Windows 操作系统以外，其余的如 Linux、UNIX 等各个版本或多或少地都存安全漏洞。虽然设计者不断地发现公布新的漏洞，但是在修改了已有的漏洞之后又将会出现新的漏洞，软件的漏洞有两类：一类是有意制造的漏洞，另一类是无意制造的漏洞。有意制造的漏洞是指设计者为日后控制系统或窃取信息而故意设计的漏洞，包括各种后门、逻辑炸弹。例如当年风靡电脑界的江民逻辑炸弹，该逻辑炸弹在特定条件下对计算机实施破坏，其结果与某些计算机病毒的破坏作用相似，可以造成电脑软硬盘都不能启动。ASP 源码问题是 IIS 服务的设计者留下的一个后门，任何人都可以使用浏览器从网络上方便地调出 ASP 程序的源码，从而可以收集系统信息，进而对系统进行攻击。无意制造的漏洞是指系统设计者由于疏忽或其他技术原因而留下的漏洞。比如：使用 C 语言的字符串复制函数，因未做合法性检查而导致缓冲区溢出。总之这些漏洞成为黑客攻击

的便利途径，所以要及时对系统和应用程序打上最新的补丁，及时更新软件。

2. 协议安全的脆弱性

计算机的运行以及网络的互联，都是基于各种通信协议的基础之上的，但是因特网设计的初衷是为了计算机之间交换信息和数据共享，缺乏对安全性整体的构想和设计，协议的开放性、复杂性以及协议在设计时缺乏认证机制和加密机制，这些使得网络安全存在着先天性的不足。当前计算机系统使用的 FTP、E-mail、NFS 以及互联网赖以生存的 TCP/IP 协议等都包含许多影响网络安全的因素，存在着许多漏洞。例如 IP 欺骗就是利用了 TCP/IP 网络协议的脆弱性。

3. 数据库管理系统安全的脆弱性

数据库主要应用于客户/服务器(C/S)平台。在 Server 端，数据库由 Server 上的 DBMS 进行管理。由于 C/S 结构允许服务器有多个客户端，各个终端对于数据的共享要求非常强烈，这就涉及数据库的安全性与可靠性问题。当前大量的信息都存储在各种各样的数据库中，然而在数据库系统安全方面考虑却很少，有时数据库管理系统的安全与操作系统的安全不配套，这就导致了数据库不安全性因素的存在，对数据库构成的威胁主要有对数据的破坏、泄漏和修改等。

4. 人员的因素

人为的无意失误，如操作员安全配置不当造成的安全漏洞，用户安全意识不强，用户口令选择不慎，用户将自己的账号随意转借他人或与别人共享等都会给网络安全带来威胁。

人为的恶意攻击，是计算机网络所面临的最大威胁，敌手的攻击和计算机犯罪就属于这一类。此类攻击又可以分为以下两种：一种是主动攻击，它以各种方式有选择地破坏信息的有效性和完整性；另一种是被动攻击，它是在不影响网络正常工作的情况下，进行截获、窃取、破译以获得重要机密信息。这两种攻击均可对计算机网络造成极大的危害，并导致机密数据的泄漏。

主要的攻击与威胁手段如下。

(1) DoS：使目标系统或网络无法提供正常服务。

DoS(Denial of Service)，也就是“拒绝服务”的意思。最基本的 DoS 攻击就是利用合理的服务请求来占用过多的服务资源，从而使合法用户无法得到服务，如图 1-1 所示。基本过程是：攻击者向服务器发送众多的带有虚假地址的请求，服务器发送回复信息后等待回传信息，由于地址是伪造的，所以服务器一直等不到回传的消息，分配给这次请求的资源就始终没有被释放。在这种反复发送伪地址请求的情况下，服务器资源最终会被耗尽。

(2) 扫描探测：系统弱点探察。

(3) 口令攻击：弱口令。

(4) 获取权限，提升权限。

猜/crack root 口令、缓冲区溢出、利用系统或应用 Bugs。

(5) 插入恶意代码：病毒、特洛伊木马(BO)、后门、恶意 Applet。

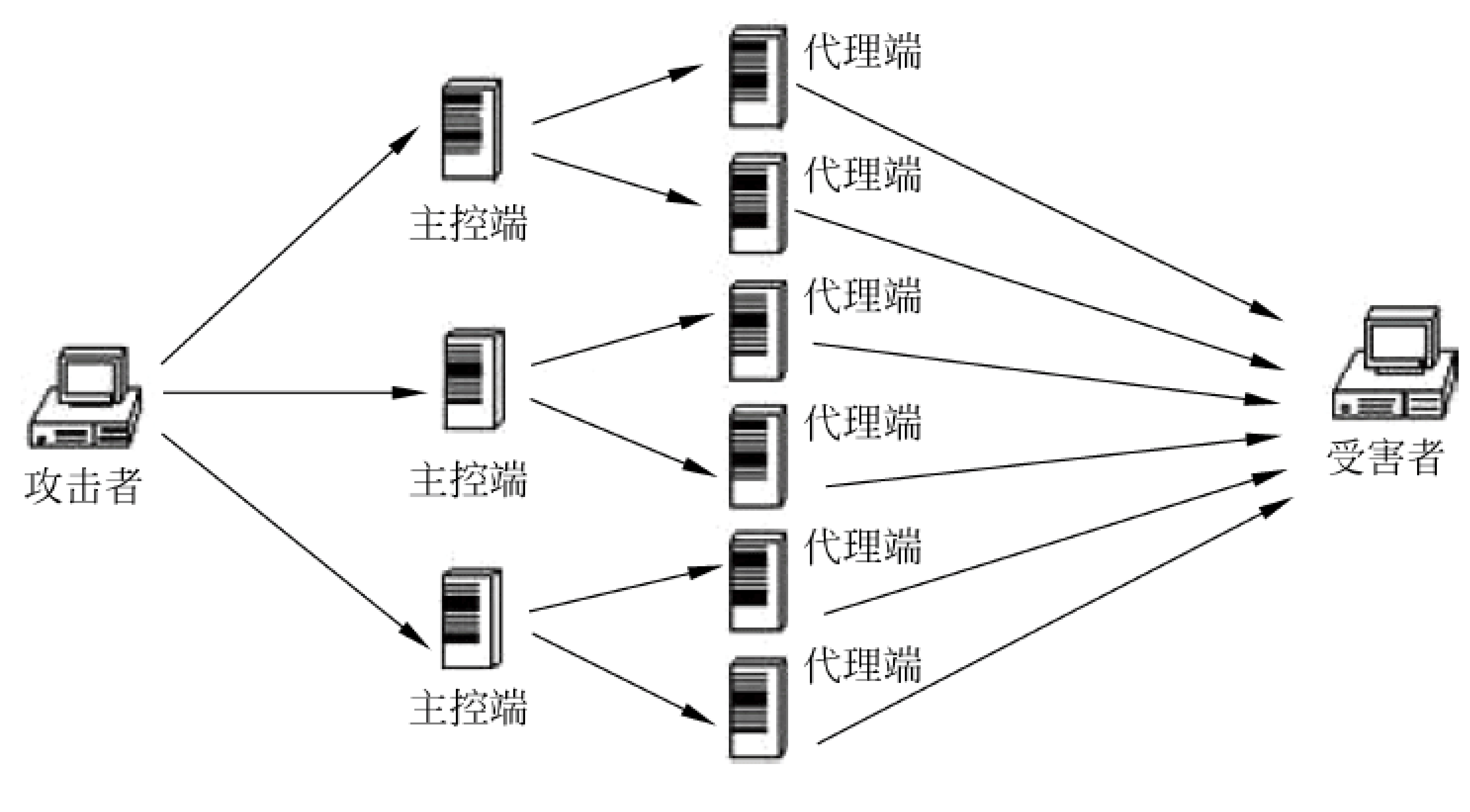


图 1-1 DoS 攻击

(6) 网络破坏：主页篡改、文件删除、毁坏 OS、格式化磁盘。

(7) 数据窃取：敏感数据拷贝、监听敏感数据传输——共享媒介/服务器监听/远程监听 RMON。

(8) 伪造、浪费与滥用资源。

(9) 篡改审计数据：删除、修改、权限改变、使审计进程失效。

(10) 安全基础攻击：防火墙、路由、账户修改，文件权限修改。

1.3 我国网络发展现状及安全趋势

1.3.1 我国互联网络的发展现状

2015 年 3 月 20 日，在工业和信息化部的指导下，由中国互联网协会、国家互联网应急中心联合主办的《中国互联网站发展状况及其安全报告(2015 年)》(以下简称《报告》)发布会暨“网站发展与安全趋势论坛”在北京举行。来自工业和信息化部电信管理局、通信保障局，国家互联网信息办公室网络社会工作局、数据技术局，北京市通信管理局等单位的有关领导出席了此次会议。中国互联网协会备案部负责人、国家互联网应急中心高级工程师何世平、优刻得云计算(UCloud)CEO 季昕华、知道创宇运行中心总监潘少华、宝利九章 CTO Jerry Kang、东方博盾董事长高振宇等专家分别在会上作了主题报告。

本次报告是国内针对中国网站发展状况及其安全的顶级、专业、权威研究报告，报告对中国网站发展总量、中国网站接入服务市场竞争情况、中国网站分布情况、网站主办者组成情况、网站所注册使用的独立域名、专业互联网信息服务网站发展、中国互联网行业创业创新和中国网站的安全状况等方面进行了全面、深入的统计、分析和研究。

报告显示，近三年来，中国网站的发展实现止跌回升、呈现出稳中求进的发展态势，在部分行业和领域培育出了一批具有产业竞争能力和一定规模的互联网企业，具体情况如下：

(1) 中国网站总量继续保持规模化发展。截至 2014 年 12 月底，中国网站总量达到 364.7 万余个，同比年度净增长约 14.1 万个，为中国网站提供互联网接入服务的接入服务商 1068 家，同比年度净增长 86 家，网站主办者近 285 万个，其中网站主办者为单位的约

211.9 万个、网站主办者为个人的 73.1 万余个，中国网站所使用的独立域名共计 481.2 万余个，每个网站主办者平均举办网站 1.3 个，每个中国网站使用的独立域名平均 1.3 个。全国提供教育、医疗保健、药品和医疗器械、新闻等专业互联网信息服务的网站约 2.1 万个。

（2）中国网站接入服务市场经营主体多元化，民营经济春色满园。一是从事网站接入服务业务的市场经营主体快速增长。因 IDC、ISP 增值电信业务许可证恢复受理，全年全国新增已从事网站接入服务业务的市场经营主体 86 家。二是国退民进，市场份额保持均衡，垄断市场尚未形成。三家基础电信企业直接接入的网站仅为中国网站总量的 7%，同比下降 2 个百分点，单一接入服务商市场份额均未超过 25%。三是接入服务市场资本多元化发展加速推进，形成国有资本、民间资本和境外资本共融共进的多种所有制经济发展新格局。四是民营接入服务商发展成就显著。接入网站数量排名前 10 的接入服务商均为民营接入服务商，接入网站数量排名前 20 的接入服务商中只有一家基础电信企业省级公司，且排名在第十五。五是云计算等新型网站接入方式发展迅猛。以 UCLOUD、腾讯云、百度云、阿里云等为代表的云计算公司如雨后春笋，不断涌现并发展壮大。

（3）中国网站区域发展不平衡。与中国经济发展高度相似，中国网站在地域分布上呈现东部地区多、中西部地区少的发展格局，区域发展不平衡的问题较为突出。无论从网站主办者住所所在地统计，还是从接入服务商接入所在地统计，网站主要分布在广东、北京、江苏、上海、浙江、福建、山东等东部沿海省市。

（4）中国网站主办者中单位举办网站是主流。在 364.7 万个网站中，网站主办者为“单位”举办的网站达到 272.6 万余个，占中国网站总量的 74.7%，同比年度净增长 27.4 万余个。其中“企业”举办网站约 252.6 万个、“事业单位”举办网站 9.3 万余个、“政府机关”举办的网站 5.9 万余个、“社会团体”举办的网站约 4.8 万个。而“个人”举办的网站约 92.1 万个，占中国网站总量的 25.2%，同比年度减少 13.4 万余个。

（5）中国网站注册使用的独立域名继续呈“三国争霸”态势。在中国网站注册使用的 481.2 万余个独立域名中，涉及的顶级域 302 个，较去年（277 个）增长 25 个，其中注册使用 .cn、.com、.net 域名的中国网站数量最多，.cn、.com、.net 独立域名使用数量占整个独立域名总量的 94.6%。截至 2014 年 12 月底，.com 域名使用数量最多，达到约 295.9 万个；其次为 .cn 和 .net 域名，各使用 129.5 万余个和约 29.6 万个。

（6）中国网站注册使用中文域名和新通用顶级域名已被市场认可。中国网站注册使用的中文域名主要有 5 个，分别为“.中国”、“.公司”、“.网络”、“.公益”、“.政务”，使用总量为 69 633 个，其中“.中国”的域名数量最多，52 785 个，占中文域名使用总量的 75.8%。与此同时，随着新通用顶级域的入根解析，为网站配置使用新通用顶级域名成为一大趋势。如“.wang”、“.商城”、“.网址”、“.我爱你”、“.集团”、“.ren”等新通用顶级域名已被越来越多的中国网站主办者注册使用。

（7）专业互联网信息服务网站发展行业聚焦，文化类网站发展一年翻番。专业互联网信息服务网站主要集中在教育、医疗保健、药品和医疗器械等行业和领域，新闻、视听节目、出版等行业和领域发展规模相对较小。

（8）中国网站发展体现以人为本、注重用户体验的意识增强。中国网站主办者注重为用户提供更好的用户体验，采用网站加速、网站安全防护等互联网新技术新业务的网站数量及规模正在快速增长。据 2014 年底的数据显示，全国共有近 16 万个网站顶级域名使用网

站加速、网站安全防护等互联网新技术新业务，从事网站加速、网站安全防护等互联网新技术新业务的网宿科技、奇虎360、知道创宇、百度网讯等互联网企业的规模也在日益扩大。

(9) 中国网站保护和促进文化多样性。据统计，中国网站在使用语言的种类上，除简体中文、繁体中文和英语之外，还使用其他语言的网站数量达到2.1万余个，其中包括法语、藏语、维吾尔语、蒙古语、哈萨克语、柯尔克孜语、西班牙语、日语、俄罗斯语等11种语言，中国网站语言的多样性，有力地促进了中国人民的对外友好交往和中国的改革开放事业。

(10) 中国互联网行业创业创新活跃频繁，经过高淘汰率的激烈竞争，在部分行业和领域成长出了一批具有国际竞争力的知名企业。在国家大力实施创新驱动发展战略的带领下，互联网领域呈现出前所未有的创业创新热情和氛围，互联网领域的创业创新正在引领新一轮科技革命和产业变革，创业创新是否成功与网站生存周期关系密切，2014年全年新开通的中国网站数量约95.2万个，平均每月新开通网站7.9万余个；全年网站主办者自行停办的中国网站81.1万余个，平均每月自行停办的网站6.7万余个。经过激烈的市场竞争和洗礼，在电商、搜索、社交、游戏、文学、旅游、安全等众多领域涌现出具有一定规模的互联网企业。

(11) 针对我国境内网站的仿冒钓鱼站点成倍增长，境外攻击、控制事件不断增加，中国网站安全问题形势依然严峻。在仿冒钓鱼上，据CNCERT监测，共有6116个境外IP地址承载了93 136个针对我国境内网站的仿冒页面，仿冒页面数量较2013年增长2.1倍；中国反钓鱼网站联盟在2014年全年共处理钓鱼网站51 198个，平均每月处理钓鱼网站4266个。在篡改和植入后门上，据CNCERT监测，境内被篡改网站36 969个，较2013年大幅增长53.8%，被植入后门的网站达到40 186个，其中位于美国的4761个IP地址通过植入后门控制了我国境内5580个网站，侵入网站数量居首位。

1.3.2 信息安全趋势和任务

2014年，中共中央网络安全和信息化领导小组成立，由此掀起了信息安全的热潮。在已经来临的2015年，无论是政策导向还是技术演进，人们都不得不关注信息安全的发展趋势，明确任务。

2014年岁末，据国外媒体报道，索尼影业遭到黑客攻击，公司办公室内的所有电脑均无法操作。索尼创建的十多个Twitter营销账号似乎也遭到了攻击，不停发送内容相同的消息。据称，黑客们已经从索尼影业获得了大量敏感文件，其中一些文件在打包压缩后被发送到了互联网上。此次攻击影响了索尼影业的日常办公。有报道称，索尼影业的员工们无法发送电子邮件，无法使用电脑，甚至无法接听电话。一名索尼影业的员工对国外媒体表示，他们处于完全瘫痪的状态中……

信息安全是永远的话题。攻击对象在变化，攻击方式在变化，攻击技术在变化，唯一不变的，就是对恶意攻击的坚决抵抗和不妥协。

2014年2月27日，中共中央网络安全和信息化领导小组第一次会议召开。中央网络安全和信息化领导小组的成立，被认为是中国开始高度重视信息安全，希望在信息安全领域有所作为的标志性事件。

由此，中国的信息安全热潮被掀起。驻足回首，人们看到，在已经走过的2014年，信息安全市场正在逐渐成长；登高远眺，人们又会发现，在已经到来的2015年，信息安全领域将

呈现更多的变化，值得关注。

互联网正在改变世界。同样，它也在改变人们对信息安全的认知。

互联网、移动互联网、物联网的不断演进，颠覆了人们的生活方式。然而对于攻击者而言，万物互联同样意味着可攻击的目标增多，攻击方法在创新，“攻击面”在迅速扩大，相对应的信息安全防御链条则越来越长，防御思想也在不断进化，信息安全防御面临很大的创新压力。

1. 各安全公司对信息安全的现状分析

360公司副总裁曲晓东认为，从索尼影业被攻击等近期频发的一系列安全事件看，现在的安全威胁已经发生了“质”的变化。“攻击者不再是某些个体，而是有组织的团队；攻击也不再是漫无目的的行为，而是瞄准了特定目标；攻击也不再是为了炫耀技术，而是为了潜伏并窃取有价值的信息；攻击者从利用已知的工具和漏洞发展到越来越多地利用未知工具、零日漏洞甚至社会工程学等综合性手段。”持类似观点的，还有山石网科产品市场总监贾彬。他表示，像索尼这样的大型企业，一定拥有比较完备的信息安全防护措施和手段，但是仍然无法避免被攻陷。这就说明，恶意攻击正在升级，给企业的安全防护带来了更大的挑战。

“真正可怕的是，‘攻’进步了，‘防’却没有跟上。”曲晓东说。

启明星辰首席战略官潘柱廷告诉记者，木马、病毒、钓鱼等常见的攻击方式仍然有效，而诸如DDoS等蔓延式的攻击也很难防范，伴随网络带宽的进一步增加，2015年甚至可能会出现1TB流量的DDoS攻击。更为重要的是，APT攻击呈现出泛化的趋势，出于商业竞争的目的，它将更多地出现在企业之间。

不可否认，无论是防病毒还是防攻击，无论是已知威胁检测还是未知威胁检测，安全似乎一直处于后知后觉的状态——发现威胁、分析威胁、形成具体的或通用的特征规则，然后才能对这个威胁进行防御。所以，在面对复杂、未知、定向攻击等高等级安全威胁时，传统安全的防护方式频频失手，缓不救急。

曲晓东认为，以前的信息安全重点强调的是边界安全，在网络的边界上设一些网关类的安全产品，像防火墙、IPS、IDS等。通过这些人为设置的“墙”把攻击挡在企业网络的外围，让它无法进入企业内部。然而随着互联网的发展，企业要想设置并充分利用这堵“墙”越来越困难。

员工的手机、平板电脑等个人终端设备在办公时可以连接到企业的网络，到机场就可以连接到机场的网络，如果这台终端在公共网络中感染了病毒和木马，再回到企业内网时，病毒和木马的攻击就有可能不经过企业边界的安全设备而进入企业内网。

其实，万物互联远不止智能手机可以随时随地接入网络这样简单。大量可联网智能设备的涌现，以及原先封闭的、与互联网物理隔离的工业控制系统逐渐采用通用协议接入互联网，都存在着巨大的安全风险。

“我们把这种现象叫‘边界模糊’或者‘边界消失’，这对传统的企业安全提出了严峻的挑战。如果企业的边界模糊了或者边界消失了，那么部署在企业网络边界的‘墙’就形同虚设。”曲晓东说。

潘柱廷指出，如何保障工业控制系统安全对用户和厂商都是个挑战。对于安全厂商而言，由于工业控制系统的封闭和不通用，使防护方案的研发成本极高。而且，工业控制系统

很容易成为高级威胁的攻击目标，因为它一旦出现安全问题，将有可能酿成严重的事故。

“毫无疑问，高级威胁将成为未来安全事件的主流。”曲晓东判断。

道高一尺，魔高一丈。在信息安全的攻防大战中，双方此消彼长的较量从来没有停止。如果安全威胁真的发生了质变，那么安全防御体系应该如何应变呢？

潘柱廷给出的答案是大数据。“哪个安全厂商率先掌握了大数据技术，哪个厂商就掌握了技术不对称的优势。”潘柱廷告诉记者，在攻防类安全实践中，核心任务就是威胁检测。防护一方的首要工作就是检测出威胁，只有将威胁及时、顺利地检测出来，才能让后续的防御工作有效进行。如果能够在企业网络中尽可能多地采集数据，把更多的异常数据纳入检测范围，并通过大数据技术进行分析，无疑会大大增加威胁检测的成功率。

当然，应用大数据技术完美实现威胁检测可以说是一个美好的愿景。但是，它需要解决的新问题就是如何把在企业网络中采集到的海量的、多维数据进行有效分析，从而提取有价值的信息。近年来，山石网科一直在强调智能安全的理念。在记者看来，所谓智能，必然要和大数据技术紧密联系在一起，以有效的大数据分析作为基础，才能实现真正的智能。

贾彬介绍，相对于基于特征匹配的传统威胁检测方式，新的智能安全应该是基于网络行为分析的。这是因为，攻击者会采取不同的攻击方式、不同的工具、不同的木马变种，特别是一些高等级的攻击者会针对攻击目标定制开发攻击工具，让攻击过程变得更加隐蔽，难以察觉。然而，任何一个网络攻击都会具有扫描、植入、传输等相对固定的攻击行为。对企业而言，如果防护系统能够对企业的正常业务行为进行学习，从而在日常监控中发现与正常业务行为不匹配的异常行为，就能更加高效地发现威胁。

曲晓东则强调，面对如今的 IT 环境，单纯依靠封堵的方法已经很难抵御网络攻击。除了大数据技术之外，为了应对企业边界模糊或边界消失带来的安全挑战，需要采用“云＋端＋边界联动”的立体化解决方案，只有“云＋端＋边界联动”的综合立体防御体系才能帮助企业灵活、快速、最大限度地减少来自高级安全威胁的影响和损失。

2. 安全协作是大势所趋

“现在，为了利益最大化，攻击方都能够联合起来。那么，作为保护企业信息安全的中坚力量，安全厂商为什么不能联合起来呢？”潘柱廷发出呼吁。

事实上，面对升级的安全威胁，除了创新的安全防御技术和策略，还有重要的一点就是协作。通常，为了保证对技术的足够投入和专业性，安全厂商往往只专注于信息安全的某一个或某几个领域，而且往往专注的安全厂商也更容易获得用户的信赖。但是，信息安全防御的链条越来越长，已经鲜有厂商能够提供覆盖全链条的安全防御解决方案，这时，安全协作就显得更加重要。

“在美国，安全厂商之间的协作司空见惯，一家很小的厂商都可以共享到大厂商的安全数据。”潘柱廷表示，厂商协作首先要落地的是安全数据的共享。目前，“启明星辰”已经开始和国内其他知名安全厂商进行安全数据共享，这对研究攻击行为，保障企业信息安全具有重要的意义。“此外，针对 DDoS 这样的攻击，人们也在呼吁成立‘反 DDoS’联盟。因为这种攻击从用户的角度很难防御，如果通过某种联盟的形式，可以从攻击发起端进行限制，将很大程度上改善目前对抗 DDoS 攻击的局面。”潘柱廷说。

“在国内信息安全产业链条中，大厂商之间的合作较少，同时充满了低水平、同质化的竞

争，从而形成利润率低、技术水平低的恶性循环。360公司进入企业信息安全领域，给产业带来了新技术、充裕的资金、开发用户体验更好产品的能力。”曲晓东认为，360公司进入企业信息安全市场后，与其他安全厂商进行了一系列合作和整合，给整个产业带来了新的机遇，也造成了鲶鱼效应。

显然，在信息安全产业中，政策和合规性是比较强的驱动力。

3. 我国信息安全厂商正在解决的问题

在过去的2014年，中共中央网络安全和信息化领导小组成立、首届世界互联网大会召开、首届国家网络安全宣传周启动。一系列事件让中国的企业和个人用户深切感到了信息安全的重要性，安全意识得到提升。信息安全已经深入到国家治理、企业经营和百姓的日常生活中，并且与国家安全息息相关。在信息安全和自主可控的呼声下，本土安全厂商正在迎来巨大的发展机遇，并茁壮成长。

工业和信息化部中国电子信息产业发展研究院信息安全研究所所长刘权告诉记者，据他们统计，中国整体安全产业规模在2014年达到了550亿元左右，同比增长约26%。

贾彬介绍，2014年山石网科在很多方面取得了重大突破。“随着用户信息安全意识的提升和对信息安全认知的成熟，他们在采购信息安全产品的时候从看重性价比逐渐过渡到更加看重产品的品质。这就给山石网科带来了更多的市场机会。”贾彬告诉记者，山石网科已经成为金融等行业用户在信息安全领域替换国外产品的首要选择之一。同时他预计，在2015年，山石网科在政府、金融等领域的业绩还将有大幅增长，甚至会翻番。就在记者截稿时，中国电信2014年IP网络安全设备集采结果全部出炉，山石网科凭借防火墙、防病毒、内容过滤等功能，以及高性能、高可靠等特性，成功中标该集采防火墙标段的全部四个标。

360公司的成绩则更加有目共睹。“360漏洞实验室被誉为东半球最强大的白帽子军团。目前，360已经成为全世界报告软件漏洞数量最多的安全软件公司之一，在这方面超过很多国外知名安全厂商。从2009年到2014年，360公司共63次因挖掘并协助修复漏洞获得微软官方公开致谢，漏洞挖掘与专业防护技术在世界居于领先地位。”曲晓东说。

可以说，360公司把互联网公司的产品设计理念和创新方式带入了信息安全领域。他们对国外新兴安全厂商Splunk、FireEye、Bit9等进行了跟踪和深入研究，进行了安全技术和安全产品的创新。曲晓东介绍，“360天眼”目前已经是一款非常成熟的产品，它是360公司针对企业用户推出的带有大数据功能的边界安全产品，可以提供全面的定向攻击、APT攻击检测与分析方案，专注于高级威胁的发现，能够对各种已知和未知威胁进行不同维度、不同攻击阶段、不同攻击技术的立体化检测，并且利用大数据、威胁情报及可视化分析等多种先进技术，回溯攻击过程、分析攻击技术和其影响范围，确定攻击目的。曲晓东认为“360天眼”是解决当前企业面临的主流安全威胁的卓有成效的安全工具。

“基于‘云+端+边界’的企业安全立体防护理念，我们还推出了针对桌面端和移动端的安全产品‘360天擎’和‘360天机’，它们成为优秀的企业终端管控系统，在政府、电力、航空、能源、金融等领域均已经有用户在使用。”曲晓东说。

就在中共中央网络安全和信息化领导小组第一次会议召开的两周之前，美国政府发布了由美国国家标准技术研究所（NIST）制定的《关键基础设施网络安全框架》。这是棱镜门事件后，美国政府首次出台国家级信息安全指导规范，也是奥巴马政府2013年启动保护关

键基础设施信息安全战略以来的第一个基础性框架文件。

政策是信息安全产业绕不开的话题，因为信息安全不仅是技术问题，更是产业问题。在中国信息安全产业蓬勃发展之际，人们应当看到与美国等科技和网络安全强国的差距，加快步伐，这也是我们的任务。

“要应对美国等西方国家的信息安全政策，我们就应该有更多的动作，尽快制定相应的政策，从政策层面上达到攻守平衡。”潘柱廷向记者表示。

中共中央网络安全和信息化领导小组的成立给予了信息安全最高的组织和机构保障，从而有效推动了信息安全的政策和法治建设。刘权向记者介绍，2014 年，中共中央网络安全和信息化领导小组办公室召开专题会议讨论网络立法问题，全国人大也将《网络安全法》列入立法工作计划。2015 年，我国网络安全法治建设进程将显著加快。他预测，在网络安全立法方面，首先会加快修订原有法律，例如，在刑法修订中增加关于网络恐怖主义的相关规定；修订互联网信息服务管理办法，针对新应用建立有效的信息内容管控手段。其次，围绕关键信息基础设施保护、跨境数据流动、信息技术产品和服务供应链安全等一系列重大问题，全国人大和相关单位将开展更深入的研究，《网络安全法》将取得阶段性成果。再次，中共中央网络安全和信息化领导小组办公室等机构将加快推进网络安全审查等法律制度建设。

“美国已经建立了《经济间谍法》、《计算机欺诈与滥用法》等法律构成的相对完备的反网络窃密法律体系，相比而言，我国在此领域的法律建设还相对落后，难以对类似起诉形成反制。此外，由于近两年网络安全形势飞速变化，新威胁层出不穷，信息数据的跨境流动、移动互联网时代网络数据和隐私保护、高级可持续性威胁背景下重要信息系统保护、政府信息安全管理、信息技术产品的安全审查、网络犯罪电子证据取证程序等方面都需要立法进行规范。”刘权告诉记者。

当然，信息安全包含了信息系统安全、网络基础设施安全、内容安全等诸多方面，在哪些领域迫切需要网络安全年立法，以及如何通过立法加以规范，仍然需要明确。

1.3.3 网络安全十大趋势

2015 年 1 月 15 日，中国电子信息产业发展研究院发布了“2015 年网络安全十大趋势”。

1. 网络空间国际军备竞赛加剧

随着网络安全威胁日益常态化、复杂化和高级化，各国加快网络空间军事力量建设，网络空间军备竞赛加剧，2015 年这一趋势将更为显著。一是西方国家继续建立或增设网络部队。美国 2014 年《防务评估报告》首次明确网络部队的建设目标，俄罗斯、以色列、日本等都在扩大网络部队规模。二是各国加强网络武器和新型网络对抗技术研发。三是各国加强网络战演习。据欧盟网络与信息安全委员会(ENISA)报告，各国开展网络演习的频率大幅提高。四是美国等西方国家通过北约组织加快构建网络军事同盟，以实现集体防御。

2. 发生有组织的大规模网络攻击

黑客组织、网络犯罪团体，甚至某些国家成为网络攻击的“新玩家”，针对政府部门，以及国防、金融、能源、航天、运输等重要行业的企业和机构，实施大规模、持续性的网络攻击行

动,窃取敏感信息数据、瘫痪或摧毁重要目标。2015 年这种有组织的大规模网络攻击将更普遍发生。一是美国更多的网络监控和网络攻击行为将遭到曝光,多个国家加快发展网络攻击能力。二是出于政治目的的黑客行动主义将更加泛滥。三是受经济利益驱使,网络犯罪团体将针对有价值目标开展更密集的网络攻击,攻击行为发生频率越来越高。

3. 移动互联网安全事件增加

移动互联网安全问题将更加突出。一是针对安卓设备的网络攻击持续增加。有机构预测,2015 年针对安卓设备的恶意软件数量将是 2014 年的 2 倍。二是移动支付将可能成为网络攻击的新目标,通过挖掘系统漏洞、制造网上银行病毒等,网络犯罪分子可能获取金融敏感信息或劫持账户。三是 BYOD(自带移动设备)的兴起将带来更多针对企业或机构内部网络的攻击。

4. 智能互联设备成为网络攻击的新目标

智能互联设备给用户带来更丰富的体验,但对黑客也具有无限的诱惑性,黑客将可能利用这些设备进行更复杂、更严重的破坏。已有安全研究者利用智能汽车、医疗可穿戴设备的安全漏洞实现对设备的远程控制,对设备使用者的人身安全构成威胁。2015 年针对物联网的攻击可能成为现实,攻击者可能对家庭路由器、智能电视和互联汽车等发起攻击,以获取敏感信息数据、采取进一步破坏行动,但大规模攻击还不会出现。

5. 工业控制系统的安全风险加大

高级可持续性攻击的目标正在从传统的 IT 系统,转向石油、天然气、航空运输等行业的工业控制系统。近几年大量的实际案例中,这种趋势越来越明显。2015 年,工业控制系统的安全风险持续加大,美国、欧盟等都在采取措施加强关键领域控制系统安全保护。而在我国,80%的关键系统都使用了相同的控制系统,由于依赖国外组件、安全意识低、持续接入互联网等原因,更容易受到攻击。

6. 发生大规模信息泄露事件

近年来,全球大规模数据泄露事件频发。2015 年大规模信息泄露事件可能再次甚至多次发生,掌握大量个人信息的政府机构、大型零售企业、金融机构、移动应用服务提供商成为信息窃取的重要目标。

7. 网络安全事件造成更大损失

网络安全事件带来的经济损失越来越严重。据美国战略和国际问题研究中心报告,网络犯罪每年给全球带来高达 4450 亿美元的经济损失。2015 年,随着网络威胁更为复杂、高级,网络安全事件将带来更大损失。一是针对关键信息基础设施的网络攻击一旦成功,将带来不可估量的影响,能够导致化工厂爆炸、火车碰撞、大面积停电等重大安全事故。二是黑客攻击手段和工具将更为强大,将可对更为复杂的信息系统实施网络攻击,从而造成更大的影响和损失。三是随着智能互联设备成为网络攻击的新目标,网络安全事件将不仅造成经济损失,还可能危害设备使用者的人身安全。

8. 网络空间国际话语权的争夺更加激烈

2015 年,围绕互联网关键资源治理、网络空间国际规则等问题,各国在网络空间国际话语权的争夺将更加激烈。一是在互联网关键资源治理上,仍然存在"国家主导"和"利益相关方主导"的治理模式之争,国际社会将积极推进互联网资源管理权的变革,以改变美国等少数国家掌控互联网关键资源的局面。二是各国对网络空间规则制定主导权的争夺将更加激烈。网络空间尚缺乏一套完善的网络空间国际规则,以美国为首的西方国家,以及俄罗斯和中国等都推出了网络空间国际规则的设想,但尚未形成共识。在这样的背景下,谁掌握了制定"游戏规则"的权利,谁就掌握了网络空间话语权和制高点,可以预见未来的争夺将更加激烈。

9. 我国信息安全产业高速发展

在一系列利好政策的刺激下,2015 年我国信息安全产业将高速增长。一方面,信息安全等 IT 企业加快并购整合,针对网络安全威胁加强技术研发,推出更加智能的信息安全设备和服务。另一方面,面对愈演愈烈的网络攻击和网络犯罪,政府、金融、能源等重要行业的信息安全需求大幅增长,带动市场的快速发展。预计,2015 年信息安全产业增长率将达到 30%。

10. 我国网络安全立法取得新进展

2015 年,我国网络安全法治建设进程将显著加快。一是适应网络安全形势需要加快修订原有法律,例如,在刑法修订中增加关于网络恐怖主义的相关规定;修订互联网信息服务管理办法,针对新应用建立有效的信息内容管控手段。二是围绕关键信息基础设施保护、跨境数据流动、信息技术产品和服务供应链安全等一系列重大问题,全国人大和相关单位开展更深入的研究,《网络安全法》取得阶段性成果。三是中共中央网络安全和信息化领导小组办公室等加快推进网络安全审查等法律制度建设。

1.4 密码学在网络信息安全中的作用

在现实世界中,安全是一个相当简单的概念。例如,房子门窗上要安装足够坚固的抗变形材料以阻止窃贼的闯入;安装报警器是阻止入侵者破门而入的进一步措施;当有人想从他人的银行账户上骗取钱款时,出纳员会要求其出示相关身份证明也是为了保证存款安全;签署商业合同时,需要双方在合同上签名以产生法律效力也是保证合同的实施安全。

在数字世界中,安全以类似的方式工作着。机密性就像大门上的锁,它可以阻止非法者闯入用户的文件夹读取用户的敏感数据或盗取钱财。数据完整性提供了一种当某些内容被修改时,可以使用户得知的机制,相当于报警器。这些思想是密码技术在保护信息安全方面所起作用的具体体现。

密码是一门古老的技术,但自密码技术诞生直至第二次世界大战结束,对于公众而言,密码技术始终处于一种未知的保密状态,常与军事、机要、间谍等工作联系在一起,让人在感到神秘之余,又有几分畏惧。信息技术的迅速发展改变了这一切,随着计算机和通信技术的

迅猛发展,大量的敏感信息常通过公共通信设施或计算机网络进行交换,特别是 Internet 的广泛应用、电子商务和电子政务的迅速发展,越来越多的个人信息需要严格保密,如银行账号、个人隐私等。正是这种对信息的机密性和真实性的需求,密码学才逐渐揭去了神秘的面纱,走进公众的日常生活中。

密码技术是实现网络信息安全的核心技术,是保护数据最重要的工具之一。通过加密变换,将可读的文件变换成不可理解的乱码,从而起到保护信息和数据的作用,它直接支持机密性、完整性和非否认性。

今天,在计算机被广泛应用的信息时代,由于计算机网络技术的迅速发展,大量信息以数字形式存放在计算机系统里,信息的传输则通过公共信道。这些计算机系统和公共信道在不设防的情况下是很脆弱的,容易受到攻击和破坏,信息的失窃不容易被发现,而后果可能是极其严重的。如何保护信息的安全成为许多人感兴趣的迫切话题,作为网络安全基础理论之一的密码学引起了人们的极大关注,吸引着越来越多的科技人员投入到密码学领域的研究之中。

密码学尽管在网络信息安全具有举足轻重的作用,但密码学绝不是确保网络信息安全的唯一工具。它也不能解决所有的安全问题。同时,密码编码与密码分析是一对矛和盾的关系,它们在发展中始终处于一种动态的平衡状态。

习题

1. 简述信息安全和网络安全的关系。
2. 针对信息安全和网络安全的主要攻击手段是什么?
3. 网络安全的十大趋势都包括什么?
4. 手机使用过程中需要注意哪些安全?

第2章 古典密码技术

古典密码是密码学发展的一个阶段，也是近代密码学产生的渊源。尽管古典密码比较简单，用手工或者简单机械就可实现加密、解密过程，但研究古典密码的原理，有助于理解、构造和分析近代密码。在计算机出现前，密码学由基于字符的密码算法构成，主要是字符之间互相代替或者是互相换位，好的密码算法是结合这两种方法的。虽然现在密码算法相对复杂，但基本原理是一致的。重要的变化是古典密码对字母进行变换，而现代密码是对比特流进行变换，实际上这只是字母表长度上的改变，从26个元素变为2个元素，加密的本质与古典密码是相同的，即代替密码和换位密码。

2.1 密码学的基本概念

密码学(cryptology)是研究密码系统或通信安全的一门科学。它主要包括两个分支，即密码编码学和密码分析学。密码编码学的主要目的是寻求保证消息保密性或认证性的方法。密码分析学的主要目的是研究加密消息的破译或消息的伪造。

采用密码技术可以隐蔽和保护需要保密的消息，使未授权者不能提取信息。这其中包含以下一些基本概念。

(1) 明文：被隐蔽的消息称做明文(plaintext)。

(2) 密文：隐蔽后的消息称做密文(ciphertext)或密报(cryptogram)。

(3) 加密：将明文变换成密文的过程称做加密(encryption)。

(4) 解密：由密文恢复出原明文的过程称做解密(decryption)。

(5) 密码员：对明文进行加密操作的人员称做密码员或加密员(cryptographer)。

(6) 加密算法：密码员在对明文进行加密时，采用的一组规则称做加密算法(Encryption Algorithm)。

(7) 接收者：传送消息的预定对象称做接收者(receiver)。

(8) 解密算法：接收者在对密文进行解密时，采用的一组规则称做解密算法(Decryption Algorithm)。

(9) 加密密钥和解密密钥：加密算法和解密算法的操作通常是在一组密钥(key)的控制下进行的，分别称为加密密钥(Encryption Key)和解密密钥(Decryption Key)。

(10) 密码体制分类：根据密钥的特点将密码体制分为对称和非对称密码体制(Symmetric Cryptosystem and Asymmetric Cryptosystem)两种。

对称密码体制又称单钥(one-key)或私钥(Private Key)或传统(classical)密码体制。

非对称密码体制又称双钥(two-key)或公钥(Public Key)密码体制。

在私钥密码体制中,加密密钥和解密密钥是一样的或者彼此之间是容易相互确定的。

在私钥密码体制中,按加密方式又将私钥密码体制分为流密码(Stream Cipher)和分组密码(Block Cipher)两种。

在流密码中将明文消息按字符逐位地进行加密。

在分组密码中将明文消息分组(每组含有多个字符),逐组地进行加密。

在公钥密码体制中,加密密钥和解密密钥不同,从一个难以推出另一个,可将加密能力和解密能力分开。

(11) 截收者:在消息传输和处理系统中,除了合法的接收者外,还有非授权者。他们通过各种办法,如搭线窃听、电磁窃听、声音窃听等来窃取机密信息,称其为截收者(eavesdropper)。

(12) 密码分析:虽然不知道系统所用的密钥,但通过分析可能从截获的密文推断出原来的明文,这一过程称做密码分析(cryptanalysis)。从事这一工作的人称做密码分析员或密码分析者(cryptanalyst)。

(13) 被动攻击:对一个密码系统采取截获密文进行分析,这类攻击称做被动攻击(Passive Attack)。

(14) 主动攻击:非法入侵者(tamper)主动向系统窜扰,采用删除、更改、增添、重放、伪造等手段向系统注入假消息,以达到损人利己的目的,这类攻击称做主动攻击(Active Attack)。

(15) Kerckholf 假设:通常假定密码分析者或敌手知道所使用的密码系统,这个假设称做 Kerckholf 假设。

当然,如果密码分析者或敌手不知道所使用的密码系统,那么破译密码是更难的,但不应该把密码系统的安全性建立在敌手不知道所使用的密码系统这个前提下。因此,在设计一个密码系统时,目的是在 Kerckholf 假设下达到安全性。

根据密码分析者破译时已具备的前提条件,通常人们将攻击类型分为 4 种:唯密文攻击(Ciphertext-only Attack)、已知明文攻击(Known Plaintext Attack)、选择明文攻击(Chosen Plaintext Attack)、选择密文攻击(Chosen Ciphertext Attack)。

① 唯密文攻击:密码分析者有一个或更多的用同一密钥加密的密文,通过对这些截获的密文进行分析得出明文或密钥。

② 已知明文攻击:除待解的密文外,密码分析者有一些明文和用同一个密钥加密这些明文所对应的密文。

③ 选择明文攻击:密码分析者可以得到所需要的任何明文所对应的密文,这些明文与待解的密文是用同一密钥加密得来的。

④ 选择密文攻击:密码分析者可得到所需要的任何密文所对应的明文,解密这些密文所使用的密钥与解密待解密文的密钥是一样的。

上述 4 种攻击类型的强度按序递增,如果一个密码系统能抵抗选择明文攻击,那么它当然能够抵抗唯密文攻击和已知明文攻击。

在了解密码学的基本概念之后,可以很容易地理解通信保密系统,如图 2-1 所示。

通信中的参与者包括:

发送者(Alice),在双方交互中合法的信息发送实体。

接收者(Bob),在双方交互中合法的信息接收实体。

分析者(Eve),破坏通信接收和发送双方正常安全通信的其他实体。

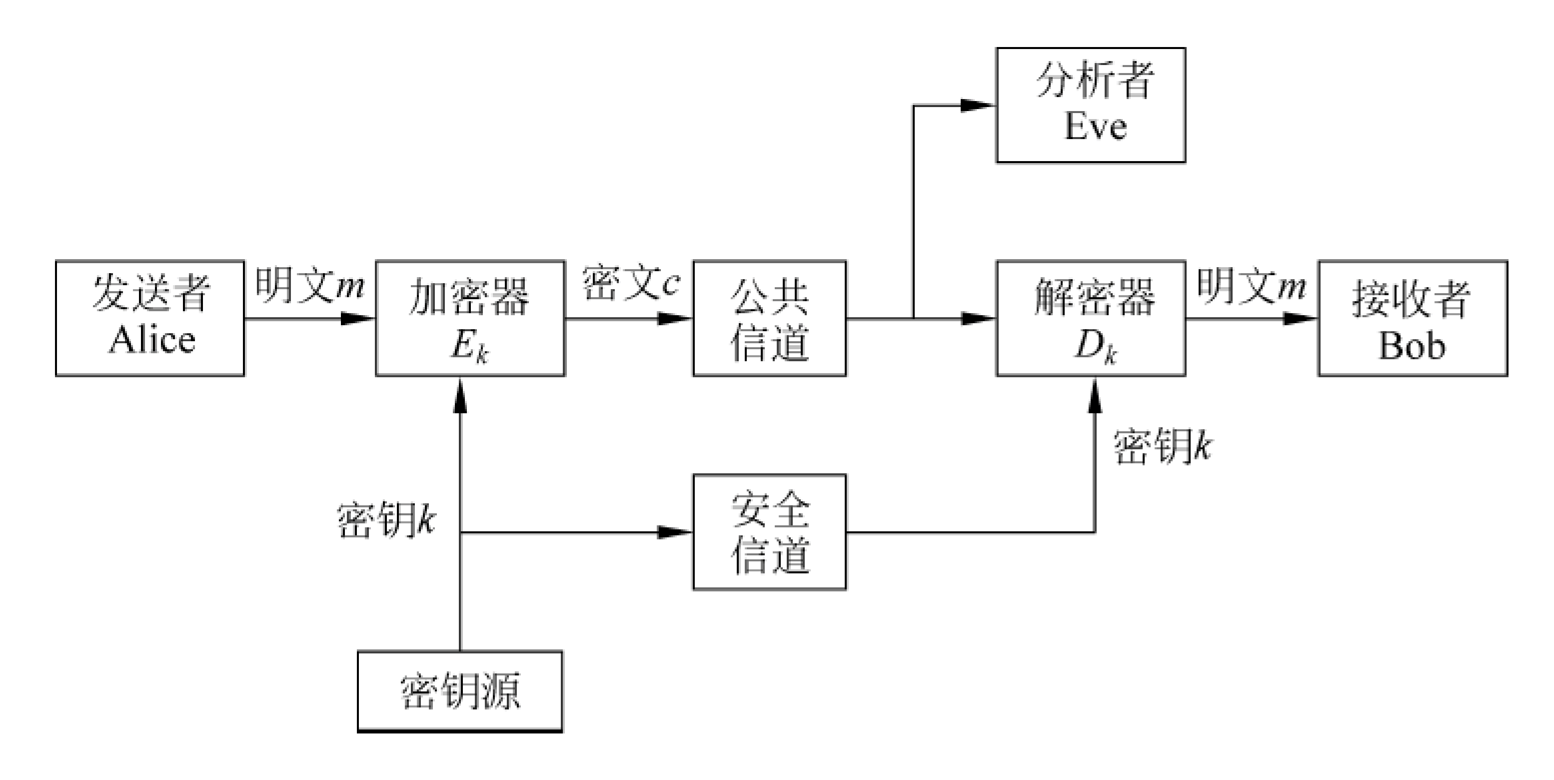


图 2-1 通信保密系统

信道,从一个实体向另一个实体传递信息的通路。

安全信道,分析者没有能力对其上的信息进行阅读、删除、修改、添加的信道。

公共信道,分析者可以任意对其上的信息进行阅读、删除、修改、添加的信道。

分析者的目的包括:

解读公共信道上的密文消息(被动)。

确定密钥以解读所有用该密钥加密的密文消息(被动)。

变更密文消息以使接收者(Bob)认为变更消息来自发送者(Alice)(主动)。

冒充密文消息发送者(Alice)与接收者(Bob)通信,以使接收者(Bob)相信消息来自真实的发送者(Alice)(主动)。

分析者常采用的方法包括中断、截获、篡改和伪造,如图 2-2 所示。

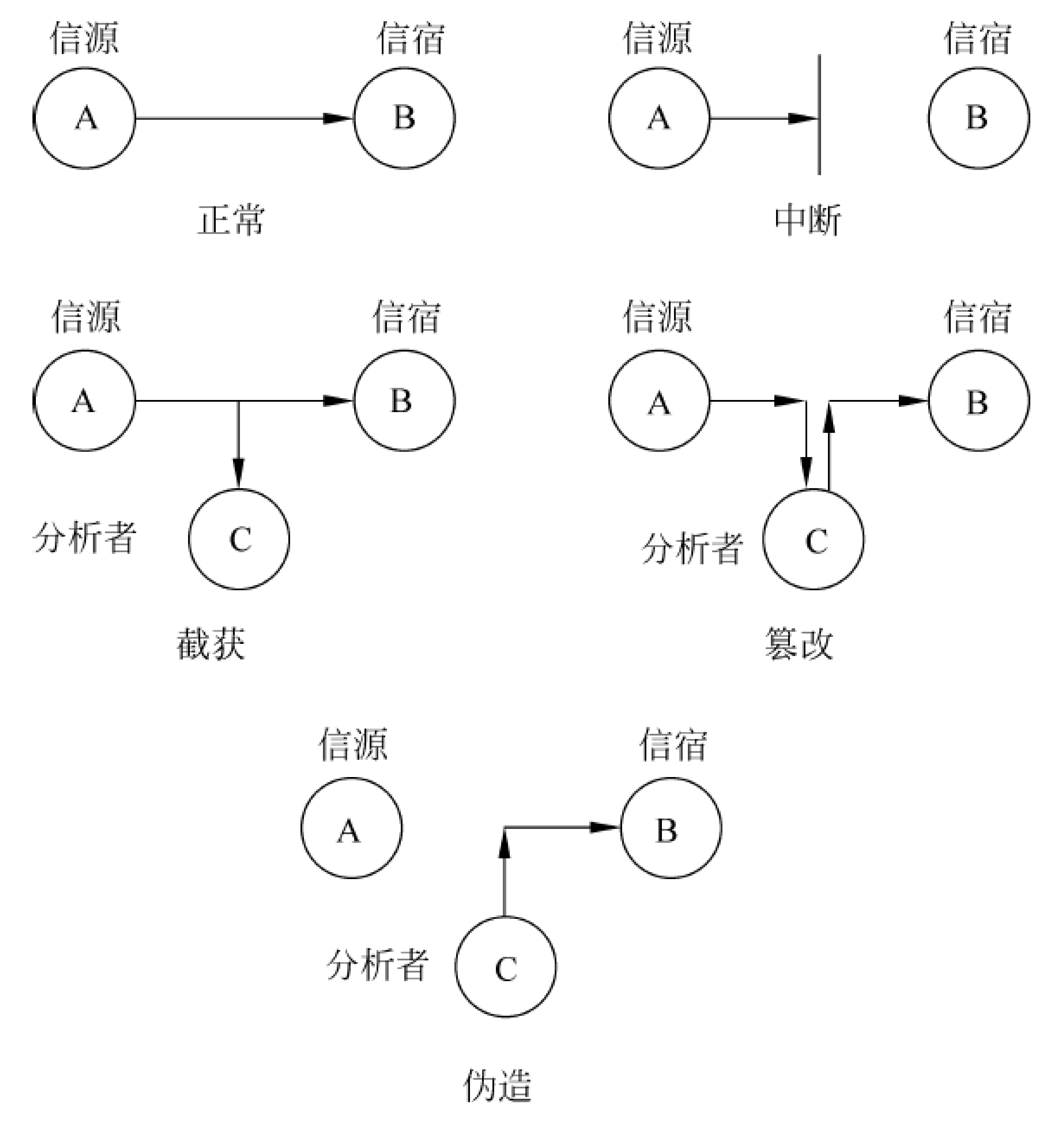


图 2-2 分析者的攻击方式

2.2 密码学的发展历史

密码学的发展历程大致经历了三个阶段：古代加密方法、古典密码方法和近代密码方法。

公元前 5 世纪，古希腊斯巴达出现原始的密码器，用一条带子缠绕在一根木棍上，沿木棍纵轴方向写好明文，解下来的带子上就只有杂乱无章的密文字母。解密者只需找到相同直径的木棍，再把带子缠上去，沿木棍纵轴方向即可读出有意义的明文。这就是最早的换位密码术。

公元前 1 世纪，著名的恺撒(Caesar)密码被用于高卢战争中，这是一种简单易行的单字母替代密码。

公元 9 世纪，阿拉伯的密码学家阿尔·金迪，同时还是天文学家、哲学家、化学家和音乐理论家。他提出频度分析方法，通过分析计算密文字符出现的频率来破译密码。

公元 16 世纪中期，意大利的数学家卡尔达诺(G. Cardano，1501—1576)发明了卡尔达诺漏格板，覆盖在密文上，可从漏格中读出明文，这是较早的一种分置式密码。

公元 16 世纪晚期，英国的菲利普斯(Philips)利用频度分析法成功破解苏格兰女王玛丽的密码信，信中策划暗杀英国女王伊丽莎白，这次解密将玛丽送上了断头台。

几乎在同一时期，法国外交官维热纳尔提出著名的维热纳尔方阵密表和维热纳尔密码(Vigenere Cypher)，这是一种多表加密的替代密码，可使阿尔·金迪和菲利普斯的频度分析法失效。

公元 1863 年，普鲁士少校卡西斯基(Kasiski)首次从关键词的长度着手将它破解。英国的查尔斯·巴贝奇(Charles Babbage)通过仔细分析编码字母的结构也将维热纳尔密码破解。

这是发生在第一次世界大战时的事情，它在世界情报学历史上占有重要地位，它使得美国举国震怒，结束中立，最终加入对德作战的行列。

第一次世界大战期间，1917 年 1 月 17 日，英军截获了一份以德国最高外交密码 0075 加密的电报。这个令人无法想象的密码系统由 1 万个词和词组组成，与 1000 个数字码群对应。密电来自德国外交部长阿瑟·齐麦曼，传送给德国驻华盛顿大使约翰·冯·贝伦朵尔夫，然后继续传给德国驻墨西哥大使亨尼希·冯·艾克哈尔特。电文将在那里解密，最后要交给墨西哥总统瓦律斯提阿诺·加汉扎。密件从柏林经美国海底电缆送到了华盛顿。英军在那里将其截获并意识到了它的重要性。英国密码破译专家开始全力以赴进行破译，然而，面对这个未曾被破译的新外交密码系统，专家们绞尽脑汁仍一筹莫展。

令英国密码破译专家意想不到的机遇降临了。接到密件的德国驻华盛顿大使约翰·冯·贝伦朵尔夫在他的华盛顿办公室里犯下了致命的错误：他们在将电报用新的 0075 密件本译出后，却又用老的密件本将电报加密后传送到墨西哥城。大使没有意识到，他已经犯下了一个密码使用者不能犯的最愚蠢、最可悲的错误。

没过多久，已经破译了老密码的英方便从德国大使的糊涂操作中获得了新旧密码的比较版本。英国的解码人员开始了艰苦的工作：将密件在旧密码中译出，用纸笔建构模型。随着齐麦曼的密件逐渐清晰，电报内容浮现出来，其重要性令人吃惊。

当时的情况是，尽管1915年美国的远洋客轮“露斯塔尼亚”号被德军击沉，但只要德国此后对其潜艇的攻击行动加以限制，美国仍将一直保持中立。齐麦曼的电文概括了德国要在1917年2月1日重新开始无限制海战以抑制英国的企图。为了让美国无暇他顾，齐麦曼建议墨西哥入侵美国，宣布得克萨斯州、新墨西哥州和亚利桑那州重新归其所有。德国还要墨西哥说服日本进攻美国，德国将提供军事和资金援助。

英国海军部急于将破译的情报通知美国，但同时又不能让德国知道其密码已被破译。于是，英国的一个特工成功地渗入了墨西哥电报局，得到了送往墨西哥总统的解了密的文件拷贝。这样，秘密就可能是由墨西哥方泄露的，它以此为掩护将情报透露给了美国。

美国愤怒了。每个美国人都被激怒了。原先只是东海岸的人在关心战局的进展，现在整个美国都开始担心墨西哥的举动。电文破译后6个星期，美国总统伍德罗·威尔逊宣布对德宣战。此时，站在他背后的是一个团结起来的愤怒的国家。齐麦曼的电文使整个美国相信德国是国家的敌人。这次破译由此也被称为密码学历史上最伟大的密码破译。

1918年，美国数学家吉尔伯特·维那姆发明一次性便笺密码，它是一种理论上绝对无法破译的加密系统，被誉为密码编码学的圣杯。但产生和分发大量随机密钥的困难使它的实际应用受到很大限制，从另一方面来说安全性也更加无法保证。

在美国Hebern发明转轮密码机的同时，欧洲的工程师们，如荷兰的Hugo Koch、德国的Arthur Scherbius都独立地提出了转轮机的概念。Arthur Scherbius于1919年设计出了历史上最著名的密码机——德国的Enigma机，在第二次世界大战期间，Enigma曾作为德国陆、海、空三军最高级的密码机，在相当长的时间内扮演着重要角色。Enigma机如图2-3所示。

Enigma机看起来是一个装满了复杂而精致的元件的盒子。不过要是把它打开来，就可以看到它可以被分解成相当简单的三个部分：键盘、转子和显示器。



图2-3 Enigma机

在图2-3的Enigma机照片上，看见水平面板的下面部分就是键盘，一共有26个键，键盘排列接近现在使用的计算机键盘。为了使消息尽量短和更难以破译，空格和标点符号都被省略。实物照片中，键盘上方就是显示器，它由标识了同样字母的26个小灯组成，当键盘上的某个键被按下时，和此字母被加密后的密文相对应的小灯就在显示器上亮起来。在显示器的上方是三个转子，它们的主要部分隐藏在面板之下。

键盘、转子和显示器由电线相连。完整的转子及转子的分解如图2-4所示。转子本身也集成了6条线路(在实物中是26条)，把键盘的信号对应到显示器不同的小灯上去。在示意图中可以看到，如果按下a键，那么灯B就会亮，这意味着a被加密成了B。同样地看到，b被加密成了A，c被加密成了D，d被加密成了F，e被加密成了E，f被加密成了C。于是如果在键盘上依次键入cafe(咖啡)，显示器上就会依次显示DBCE。这是最简单的加密方法之一，把每一个字母都按一一对应的方法替换为另一个字母，这样的加密方式是“简单替换

密码”。

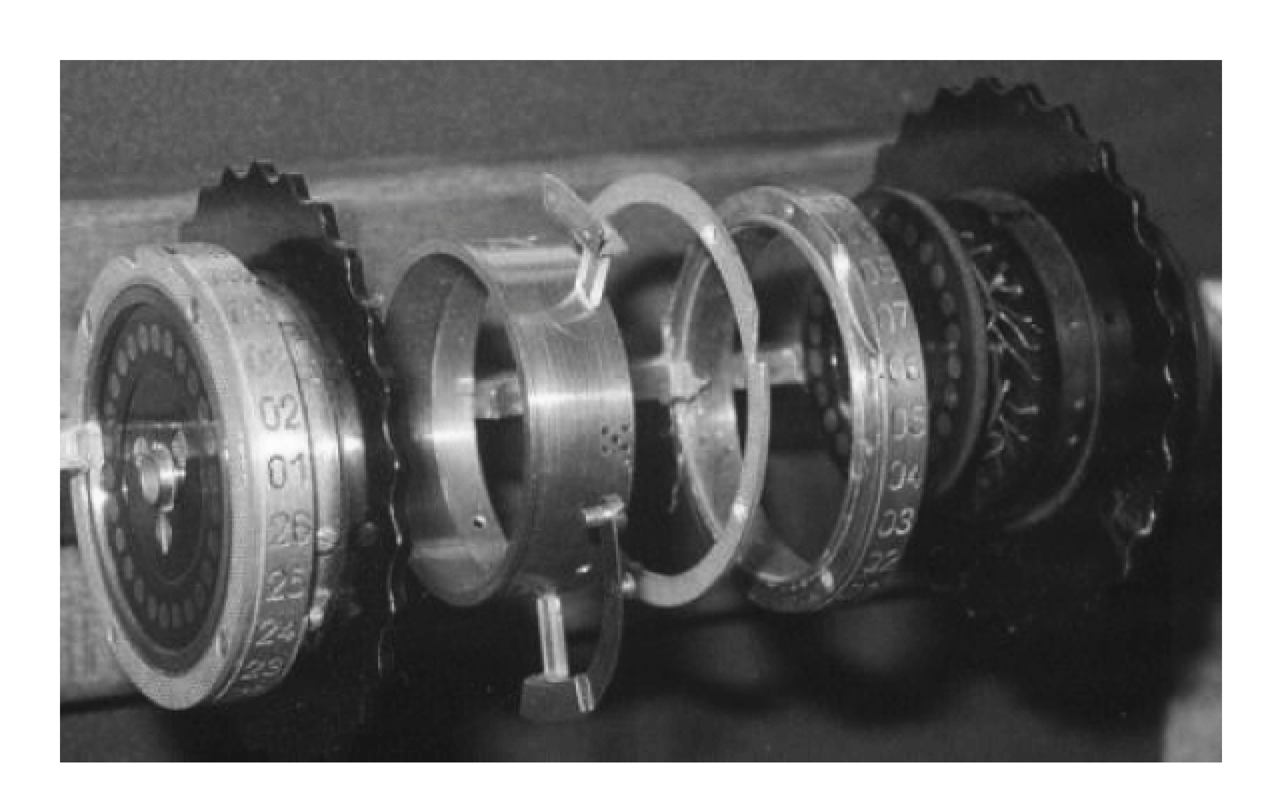


图 2-4　左部是完整的转子,右部是转子的分解

简单替换密码在历史上很早就出现了。著名的“恺撒法”就是一种简单替换法,它把每个字母和它在字母表中后若干个位置中的那个字母相对应。比如说取后三个位置,那么字母的一一对应就如下所示。

明码字母表:abcdefghijklmnopqrstuvwxyz;

密码字母表:DEFGHIJKLMNOPQRSTUVWXYZABC。

于是就可以从明文得到密文。

(veni,vidi,vici,“我来,我见,我征服”是儒勒·恺撒征服本都王法那西斯后向罗马元老院宣告的名言。)

明文:veni,vidi,vici;

密文:YHAL,YLGL,YLFL。

很明显,这种简单的方法只有 26 种可能性,不足以实际应用。一般是规定一个比较随意的一一对应,比如

明码字母表:abcdefghijklmnopqrstuvwxyz;

密码字母表:JQKLZNDOWECPAHRBSMYITUGVXF。

甚至可以自己定义一个密码字母图形而不采用拉丁字母。但是用这种方法所得到的密文还是相当容易被破解的。在公元 9 世纪,阿拉伯的密码破译专家就已经娴熟地掌握了用统计字母出现频率的方法来击破简单替换密码。破解的原理很简单:在每种拼音文字语言中,每个字母出现的频率并不相同,比如说在英语中,e 出现的次数就要大大高于其他字母。所以如果取得了足够多的密文,通过统计每个字母出现的频率,就可以猜出密码中的一个字母对应于明码中哪个字母(当然还要通过揣摩上下文等基本密码破译手段)。柯南·道尔在他著名的福尔摩斯探案集的《跳舞的人》里详细叙述了福尔摩斯使用频率统计法破译跳舞人形密码的过程。

所以如果转子的作用仅仅是把一个字母换成另一个字母,那就没有太大的应用价值了。所谓的“转子”,它会转动!这就是 Enigma 最重要的设计。当键盘上的一个键被按下时,相应的密文会在显示器上显示,然后转子的方向就自动地转动一个字母的位置。

事实上 Enigma 里有三个转子(第二次世界大战后期德国海军用的 Enigma 甚至有四个转子)。想象一下要用 Enigma 发送一条消息。发信人首先要调节三个转子的方向,使它们

处于 17 576 个方向中的一个(事实上转子的初始方向就是密匙,这是收发双方必须预先约定好的),然后依次输入明文,并把闪亮的字母依次记下来,然后就可以把加密后的消息用如电报的方式发送出去。当收信方收到电文后,使用一台相同的 Enigma,按照原来的约定,把转子的方向调整到和发信方相同的初始方向上,然后依次输入收到的密文,并把闪亮的字母依次记下来,就得到了明文。于是加密和解密的过程就是完全一样的,这都是反射器起的作用。反射器如图 2-5 所示。

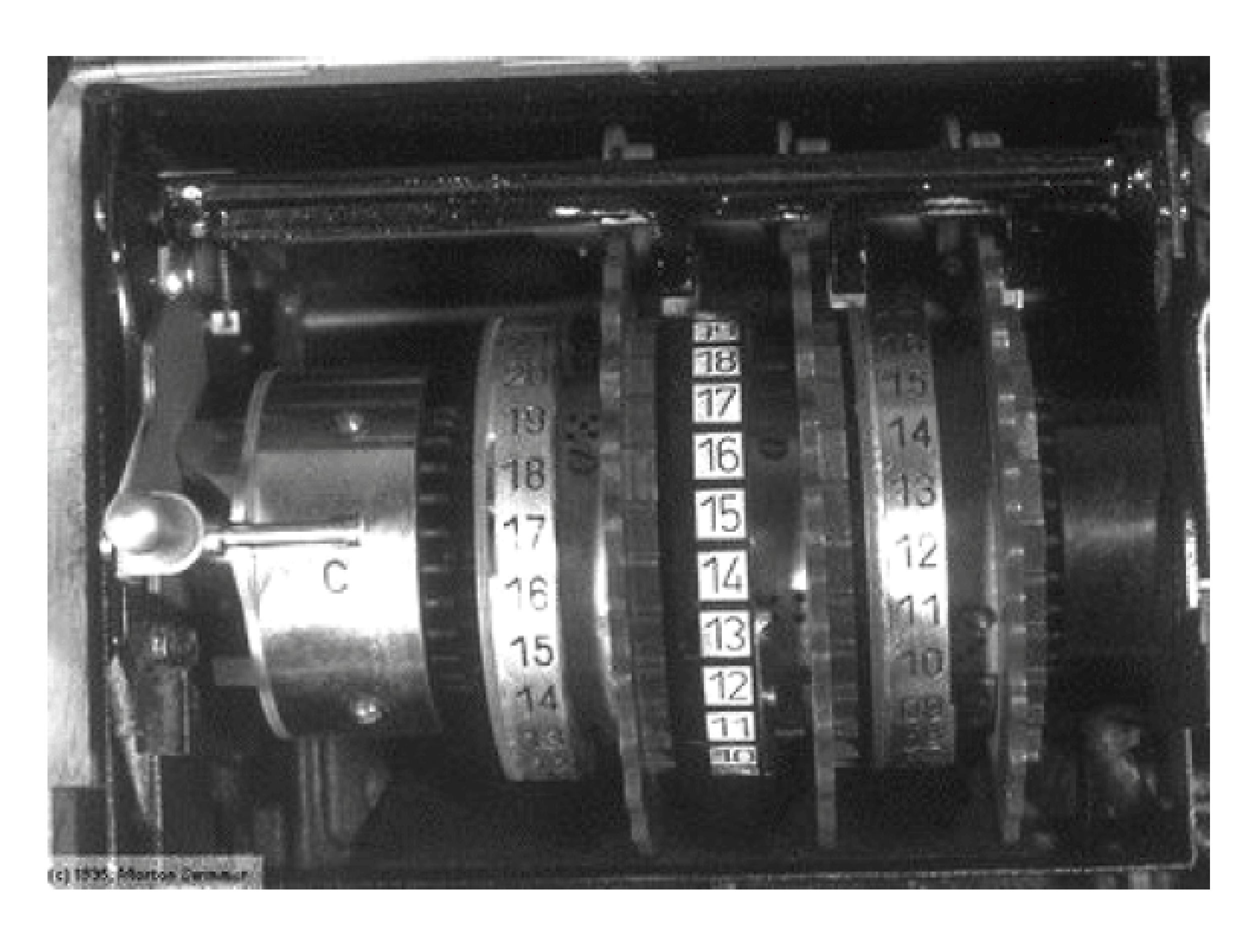


图 2-5 安装在 Enigma 中的反射器和三个转子

于是转子的初始方向决定了整个密文的加密方式。如果通信当中有敌人监听,他会收到完整的密文,但是由于不知道三个转子的初始方向,他就不得不一个个方向地试验来找到这个密匙。问题在于 17 576 个初始方向这个数目并不是太大。如果试图破译密文的人把转子调整到某一方向,然后输入密文开始的一段,看看输出是否是有意义的信息。如果不是,那就再试转子的下一个初始方向。如果试一个方向大约要一分钟,二十四小时日夜工作,那么在大约两星期里就可以找遍转子所有可能的初始方向。如果对手用许多台机器同时破译,那么所需要的时间就会大大缩短。这种保密程度是不太足够的。

当然还可以再多加转子,但是看见每加一个转子初始方向的可能性只是乘以了 26。尤其是,增加转子会增加 Enigma 的体积和成本。然而这种加密机器必须是要便于携带的(事实上它最终的尺寸是 34cm×28cm×15cm),而不是一个具有十几个转子的庞然大物。在 Enigma 的设计当中,机器的三个转子是可以拆卸下来互相交换的,这样一来初始方向的可能性就变成了原来的 6 倍。假设三个转子的编号为 1、2、3,那么它们可以被放成 123-132-213-231-312-321 6 种不同的位置,当然现在收发消息的双方除了要预先约定转子自身的初始方向,还要约定好在这 6 种排列中使用哪一种。

其次,键盘和第一转子之间还设计了一个连接板。这块连接板允许使用者用一根连线把某个字母和另一个字母连接起来,这样这个字母的信号在进入转子之前就会转变为另一个字母的信号。这种连线最多可以有 6 根(后期的 Enigma 具有更多的连线),这样就可以使 6 对字母的信号互换,其他没有插上连线的字母保持不变。在上面 Enigma 的实物图里,看见这个连接板处于键盘的下方。当然连接板上的连线状况也是收发信息的双方需要预先约定的。

于是转子自身的初始方向，转子之间的相互位置，以及连接板连线的状况就组成了所有可能的密匙，让我们来算一算一共到底有多少种。

三个转子不同的方向组成了 26×26×26=17 576 种不同的可能性；

三个转子间不同的相对位置为 6 种可能性；

连接板上两两交换 6 对字母的可能性数目巨大，有 100 391 791 500 种；

于是一共有 17 576×6×100 391 791 500，大约为 10^{16}，即一亿亿种可能性。

只要约定好上面所说的密匙，收发双方利用 Enigma 就可以十分容易地进行加密和解密。但是如果不知道密匙，在这巨大的可能性面前，一一尝试来试图找出密钥是完全没有可能的。看见连接板对可能性的增加贡献最大，那么为什么要那么麻烦地设计转子之类的东西呢？原因在于连接板本身其实就是一个简单替换密码系统，在整个加密过程中，连接是固定的，所以单使用它是十分容易用频率分析法来破译的。转子系统虽然提供的可能性不多，但是在加密过程中它们不停地转动，使整个系统变成了复式替换系统，频率分析法对它再也无能为力，与此同时，连接板却使得可能性数目大大增加，使得暴力破译法（即一个一个尝试所有可能性的方法）望而却步。

第二次世界大战中，在破译德国著名的 Enigma 机过程中，原本是以语言学家和人文学者为主的解码团队中加入了数学家和科学家。电脑之父阿兰·图灵（Alan Turing）就是在这个时候加入解码队伍的，他发明了一套更高明的解码方法，并成功解密，使德国的许多重大军事行动对盟军都不成为秘密。与此同时，美国人破译了被称为"紫密"的日本"九七式"密码机密码，并成功炸死了偷袭珍珠港的元凶——日本舰队总司令山本五十六。

在近代密码学历史上，1975 年 1 月 15 日，对计算机系统和网络进行加密的数据加密标准（Data Encryption Standard，DES）由美国国家标准局颁布为国家标准，这是密码术历史上一个具有里程碑意义的事件。

1976 年，当时在美国斯坦福大学的迪菲（Diffie）和赫尔曼（Hellman）两人提出了公开密钥密码的新思想（论文 *New Direction in Cryptography*），把密钥分为加密的公钥和解密的私钥，这是密码学的一场革命。

1977 年，美国的里维斯特（Ronald Rivest）、沙米尔（Adi Shamir）和阿德勒曼（Len Adleman）提出第一个较完善的公钥密码体制——RSA 体制，这是一种建立在大数因子分解基础上的算法。

1985 年，英国牛津大学物理学家戴维·多伊奇（David Deutsch）提出量子计算机的初步设想，这种计算机可在 30 秒钟内完成传统计算机要花上 100 亿年才能完成的大数因子分解，从而破解 RSA 运用这个大数产生公钥来加密的信息。同一年，美国的贝内特（Bennet）根据他关于量子密码术的协议，在实验室第一次实现了量子密码加密信息的通信。尽管通信距离只有 30 厘米，但它证明了量子密码术的实用性。

2.3　代替密码

代替，就是明文中的字母由其他字母、数字或符号所取代的一种方法，具体的代替方案称为密钥。代替分为单表代替密码和多表代替密码。

2.3.1 单表代替密码

公元前51年初，深冬。高卢，毕布拉克德（现法国境内伯夫雷山），恺撒的营帐。深夜，罗马共和国高卢行省长恺撒，正在一张羊皮上写着什么。他的身影被跳动的灯火映在帐篷上，高大而摇曳。他的脸略显狭长，但棱角分明，专注的神色中透着与生俱来的自负。他在写他的“随记”，也就是后来流传于世的《高卢战记》。戎马生涯的恺撒本没有余暇来写什么随记，但是过去的几年中，与他在高卢的显赫战绩相比，政治上的事态发展可不那么如意。罗马执政官克拉苏斯在同帕尔提亚人（在今土库曼斯坦南部和伊朗东北部）的作战中被俘。熔化了的台液灌进了他的喉咙……这个当年残酷镇压斯巴达克斯起义的刽子手，如今与他嗜如生命的黄金铸在了一起，这对恺撒来说是一件好事，但更是一件坏事——罗马“三巨头”之间的平衡被打破了，活着的两巨头，他和庞培，不得不面临决斗。恺撒从来没有看得起过克拉苏斯。

这个只会献媚的小人死不足惜，但庞培绝不能小看，不然的话，恺撒当年也不会把自己的女儿尤丽娅嫁给庞培。要知道，庞培比恺撒还要大8岁。现在，尤丽娅已经去世，他们之间除了你死我活已无任何瓜葛。庞培以罗马唯一执政官的优势地位，正在元老院里向他发动强大的政治攻势……他必须宣传自己，必须向元老院陈述自己的功绩，但同时又必须表现出一种谦逊、客观的态度，不能带有任何自吹自擂的痕迹。为此，他在这部随记中，处处用第三人称称呼自己，通篇都用异常平静、简洁的笔调叙说战事的经过。

这时他正写到卷五，说的是公元前54年，他的爱将西塞罗突然遭到维尔纳人的围攻，情况紧急，“于是，他以极大的酬报说服了一个高卢骑兵，送一封信去给西塞罗。送去的信是用希腊文写的，免得它被敌人截住后得知我军的计划……”写到这里，他停了一下，似在考虑更好的措词。一丝狡猾的微笑从他脸上掠过，他继续写了下去……时间无情地飞驰，转眼就过了近2000年，恺撒的《高卢战记》以其翔实的叙事、清纯的文风，成为研究罗马历史、拉丁文学和军事史不可或缺的学术资料。

1979年，中国商务印书馆将《高卢战记》译成中文，作为“汉译世界学术名著丛书”中的一种出版，译者任炳湘先生打开这本中文译本，翻到第124页，看到了上面引述的那桩派人送信给西塞罗的事。然而，治学严谨的译者在这里发现了问题，他注道：“言下之意，似乎高卢人个个不懂希腊语，即令书信被截去，也不会泄露自己的计划。但在本书卷25节中曾说到在厄尔维儿人营中发现用希腊文写的统计数字，又说高卢人无论公私文件都用希腊文书写，似乎有矛盾。”对此，译者的推测是：“也许上面两节指的是高卢人用希腊字母书写自己的语言，这一节所说的是真正的希腊文。”译者的质疑可说是切中要害，然而译者的推测却仍让人疑云难消。敌营中就没有一人认识真正的希腊文？他们就不能去找一个希腊人来识这封信（如果他们截住了这封信的话）？足智多谋的恺撒会不考虑这些明摆着的可能而铤而走险？是不是可以有另外的解释？确实有另外一种解释：如果让一位密码学家来进行推测，他会毫不犹豫地认为，恺撒送去的这封信是用密码写的！因为任何一本讲述密码学历史的著作，都会提到恺撒对军事密码学的贡献。恺撒在其军事行动中使用了密码，这在密码学界已不是密码。

现在已经无法弄清恺撒密码在当时有多大的效果，但是有理由相信它是安全的。因为恺撒的大部分敌人都是目不识丁的，而其余的则可能将这些消息当作某个未知的外语。即

使有某个敌人获取了恺撒的加密信息，根据现有的记载，当时也没有任何技术能够解决这一最基本、最简单的替换密码。现存最早的破解方法记载在公元 9 世纪阿拉伯的阿尔·肯迪有关发现频率分析的著作中。

恺撒系统的密码是自己选的一个单词。

例如，选用 mountain，写出以下的字母序列：mountaibcdefghjklpqrsvwxyz。

就是在正常字母序列中抽掉密码 mountain。由于 mountain 中有两个 n，把第二个去掉。

然后，把正常字母序列写在这个序列下面：

Mountaibcdefghjklpqrsvwxyz……密文字母序

Abcdefghijklmnopqrstuvwxyz……明文字母序

在加密的时候，用上面那个序列里的字母代替原文中的字母写成密文。例如，m 代替 a，o 代替 b。解密时方向相反。所以，加密 heishere 的结果是：btcqbkpt。

恺撒密码是单表代替密码的经典算法。设明文为 x、密文为 y、加密变换是 e、解密变换是 d。26 个字母中 a 用数字 0 代替、z 用数字 25 代替，不区分大小写，那么恺撒密码可以表示为

加密：$y=e(x) = (x+3) \bmod 26$

解密：$x=d(y) = (y+26-3) \bmod 26$

例 2-1　明文为 China，用恺撒密码求密文。

解：China 中的 5 个字母对应的数字分别为 2、7、8、13、0，所以

$y(1)=e(x)=(x+3) \bmod 26=(2+3) \bmod 26=5$

$y(2)=10$

$y(3)=11$

$y(4)=16$

$y(5)=3$

表示分别对应的字母，即密文为 f、k、l、q、d。

恺撒密码中，一个字母加密后对应的是同一个字母，如果经常使用，很容易被破译，所以可以通过修改密钥值来增加安全性，即用密钥 k 来代替 3，使其是一个变化的密钥。这种变化称为通用恺撒密码。

加密：$y=e(x)=(x+k) \bmod 26$

解密：$x=d(y) = (y+26-k) \bmod 26$

单表代替密码的缺点是密钥较小，不能抵抗穷尽搜索攻击，即便密钥值是变化的，最大也只有 26。同时，单表代替密码也不能抵抗频率分析的攻击。所谓频率分析即指根据英文单词中字母出现的频率来确定明文。字母频率表如表 2-1 所示，频率图如图 2-6 所示。

从表 2-1 中可以得出

高频字母：E、T、A、O、N、I、R、S、H；

中频字母：D、L、U、C、M；

低频字母：P、F、Y、W、G、B、Y(V?)；

稀频字母：J、K、Q、X、Z。

如果在大量的密文中，出现某个字母的次数最多，那么它的明文极有可能是字母 E。

表 2-1 字母频率表

字母	概率	字母	概率	字母	概率	字母	概率
A	0.082	H	0.061	O	0.075	U	0.028
B	0.015	I	0.070	P	0.019	V	0.010
C	0.028	J	0.002	Q	0.001	W	0.023
D	0.043	K	0.008	R	0.060	X	0.001
E	0.127	L	0.040	S	0.063	Y	0.020
F	0.022	M	0.024	T	0.091	Z	0.001
G	0.020	N	0.067				

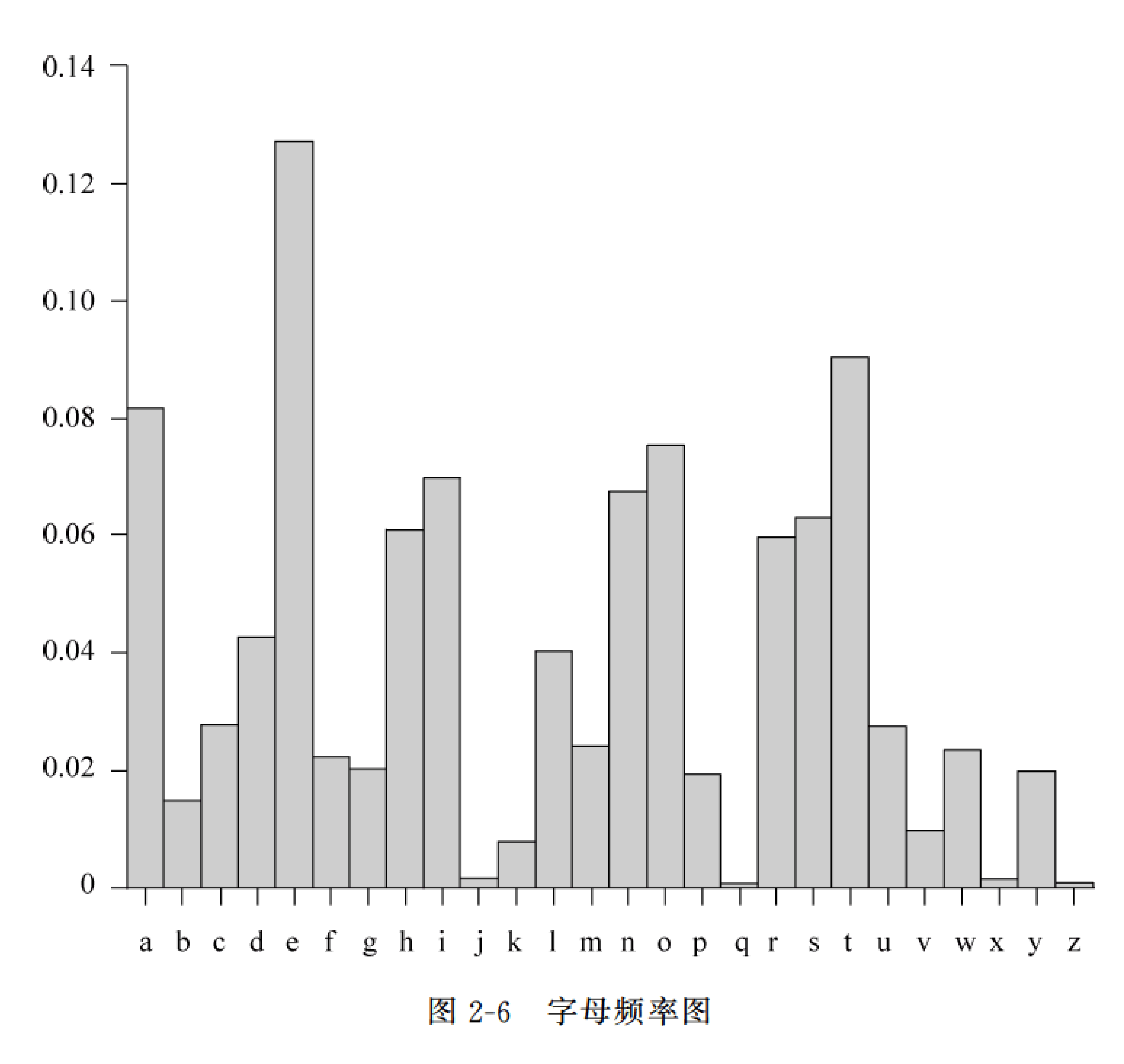


图 2-6 字母频率图

2.3.2 多表代替密码——Playfair 密码

Playfair 密码是多表代替密码的经典算法。Playfair 密码出现于 1854 年，由 Charles Wheatstone 发明，它将明文中的双字母组合作为一个单元对待，并将这些单元转换为密文双字母组合。下面介绍具体的加密方法。

1. 构造矩阵

Playfair 密码基于一个 5×5 字母矩阵，该矩阵使用一个关键词(密钥)来构造，其构造方法是：从左至右、从上至下依次填入关键词的字母(去除重复的字母)，然后再以字母表顺序依次填入其他字母。字母 I 和 J 被算为一个字母(即 J 被当作 I 处理)。

2．明文分组

将明文字符串按两个字母一组进行分组，分组之后，如果相邻两个字母相同，则在它们之间插入一个字符（事先约定的字母，如 Q）；如果明文字母数为奇数，同样要在明文的末端添加某个事先约定的字母作为填充。

3．加密方法

对每一对明文字母 P_1、P_2 的加密方法如下：

① 若 P_1、P_2 在同一行，则对应的密文 C_1 和 C_2 分别是紧靠 P_1、P_2 右端的字母。其中第一列被看作最后一列的右方（解密时反向）。

② 若 P_1、P_2 在同一列，则对应的密文 C_1 和 C_2 分别是紧靠 P_1、P_2 下方的字母。其中第一行看作最后一行的下方（解密时反向）。

③ 若 P_1、P_2 不在同一行，也不在同一列，则 C_1 和 C_2 是由 P_1 和 P_2 确定的矩形的其他两角的字母，并且 C_1 和 P_1、C_2 和 P_2 同行（解密时处理方法相同）。

例 2-2 明文为 very good，密钥为 fivestar，用 Playfair 密码求密文。

解：（1）构造矩阵。

$$\begin{bmatrix} a & b & c & d & e \\ f & g & h & i & k \\ l & m & n & o & p \\ q & r & s & t & u \\ v & w & x & y & z \end{bmatrix} \xrightarrow{\text{fivestar}} \begin{bmatrix} f & i & v & e & s \\ t & a & r & b & c \\ d & g & h & k & l \\ m & n & o & p & q \\ u & w & x & y & z \end{bmatrix}$$

（2）分组。

明文：very good

ve ry go qo dq

（3）加密。

ve：同行，所以密文为 es；

ry：对角线，所以密文为 bx；

go：对角线，所以密文为 hn；

qo：同行，所以密文为 mp；

dq：对角线，所以密文为 lm。

即密文为 es bx hn mp lm。

例 2-3 明文为 information security，密钥为 fivestar，用 Playfair 密码求密文。

解：（1）构造矩阵。

$$\begin{bmatrix} a & b & c & d & e \\ f & g & h & i & k \\ l & m & n & o & p \\ q & r & s & t & u \\ v & w & x & y & z \end{bmatrix} \xrightarrow{\text{fivestar}} \begin{bmatrix} f & i & v & e & s \\ t & a & r & b & c \\ d & g & h & k & l \\ m & n & o & p & q \\ u & w & x & y & z \end{bmatrix}$$

(2) 分组。

明文：in fo rm at io ns ec ur it yq

ve ry go qo dq

(3) 加密。

in：同列，所以密文为 aw；

fo：对角线，所以密文为 vm；

rm：对角线，所以密文为 to；

at：同行，所以密文为 ra；

io：对角线，所以密文为 vn；

ns：对角线，所以密文为 qi；

ec：对角线，所以密文为 sb；

ur：对角线，所以密文为 st；

it：对角线，所以密文为 fa；

yq：对角线，所以密文为 zp。

即密文为 aw vm to ra vn qi sb st fa zp。

例 2-4 密文为 very good，密钥为 fivestar，用 Playfair 密码求明文。

解：(1) 构造矩阵。

$$\begin{bmatrix} a & b & c & d & e \\ f & g & h & i & k \\ l & m & n & o & p \\ q & r & s & t & u \\ v & w & x & y & z \end{bmatrix} \xrightarrow{\text{fivestar}} \begin{bmatrix} f & i & v & e & s \\ t & a & r & b & c \\ d & g & h & k & l \\ m & n & o & p & q \\ u & w & x & y & z \end{bmatrix}$$

(2) 分组。

密文：very good

ve ry go od

(3) 解密。

ve：同行，所以明文为 iv；

ry：对角线，所以明文为 bx；

go：对角线，所以明文为 hn；

od：对角线，所以明文为 mh。

即明文为 iv bx hn mh。

在解密时，需要根据对单词的识别来判断明文的真实含义，如果解密的明文里含有预先约定的字母 Q(q)，需要人工去判断是否为真实的明文，这也是 Playfair 密码的缺点之一。

2.3.3 多表代替密码——Vigenere 密码

Vigenere 密码是由法国密码学家 Blaise de Vigenere 于 1858 年提出的一种密码，它是一种以移位代换为基础的周期代换密码。

设 m 是一个整数。定义 $P=C=K=(Z_{26})^m$。对任意的密钥 $K=(k_1,k_2,\cdots,k_m)$，定义

$$e_k(x_1, x_2, \cdots, x_m) = (x_1 + k_1, x_2 + k_2, \cdots, x_m + k_m) \bmod 26$$

$$d_k(y_1, y_2, \cdots, y_m) = (y_1 - k_1, y_2 - k_2, \cdots, y_m - k_m) \bmod 26$$

加密的实质如表 2-2 所示，可以将行或列的其中一个作为明文，另一个作为密钥，通过查询表就能得到相应的密文。其实 Vigenere 密码和通用恺撒密码的原理是一样的，只不过密钥不是单一的字母，而是一个字符串，如果明文字母个数大于密钥字母个数，则密钥需重复使用。

表 2-2　Vigenere 密码表

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

例 2-5　设 $m=6$，$K=$GOOGLE，明文串：BUYYOUTUBE，用 Vigenere 密码求密文。

解：密钥 GOOGLE 对应的数字分别为 6、14、14、6、11、4。

明文 BUYYOUTUBE 对应的数字分别为 1、20、24、24、14、20、19、20、1、4。

根据加密公式得

$$\begin{aligned} e_k(x_1, x_2, \cdots, x_m) &= (x_1 + k_1, x_2 + k_2, \cdots, x_m + k_m) \bmod 26 \\ &= (1+6, 20+14, 24+14, 24+6, 14+11, 20+4, 19+6, \\ &\quad\ 20+14, 1+14, 4+6) \bmod 26 \\ &= (7, 8, 12, 4, 25, 24, 25, 8, 15, 10) \end{aligned}$$

对应的字母分别为 H、I、M、E、Z、Y、Z、I、P、K，即密文为 HIMEZYZIPK。

通过查表 2-2 也能得到相同的答案。

在多表代替密码下，原来明文中的统计特性通过多个表的平均作用而被隐蔽了起来。多表代换密码的破译要比单表代替密码的破译难得多。但由于其还是相对简单，可以通过分析密文中字母重复的情况来确定多表代替密码的准确周期，即密钥字母的个数，再通过频率分析法来破译。

例如密钥：dog，明文：to be or not to be，用 Vigenere 密码加密的密文为

明文	t	o	b	e	o	r	n	o	t	t	o	b	e
密钥	d	o	g	d	o	g	d	o	g	d	o	g	d
密文	**w**	**c**	**h**	**h**	c	x	q	c	z	**w**	**c**	**h**	**h**

在密文中字符串 wchh 重复了两次，其间为 9 个字符。这个距离是密钥长度的倍数，可能的密钥长度是 1，3，9。判定了长度之后就可以用频率分析法来破译单表代替密码了。多表代替密码的破解原因主要是密钥的长度太短。

2.3.4 多表代替密码——Vernam 密码

美国电话电报公司的 Gilbert Vernam 在 1917 年为电报通信设计了一种简单方便的密码，即 Vernam 密码。

加密原理是将明文和密钥分别变换为二进制字符流，然后将明文流和密文流对位进行异或处理得到密文。

例 2-6 明文 $P=01100001$，密钥 $K=01001110$，用 Vernam 密码求密文。

解：密文 $C=P\oplus K$，即

$$\begin{array}{rr} & 01100001 \\ \oplus & 01001110 \\ \hline & 00101111 \end{array}$$

密文为 00101111。

解密 Vernam 密码用公式 $P=C\oplus K$ 即可实现。

2.3.5 多表代替密码——Hill 密码

Hill 体制是 1929 年由 Lester S. Hill 发明的，它实际上就是利用了人们熟知的线性变换方法，是在 Z_{26} 上进行的。Hill 体制的基本思想是将 n 个明文字母通过线性变换转化为 n 个密文字母，解密时只需做一次逆变换即可，密钥就是变换矩阵。

设明文 $\boldsymbol{m}=(m_1+m_2,\cdots,m_n)\in Z_{26}^n$，密文 $\boldsymbol{c}=(c_1,c_2,\cdots,c_n)\in Z_{26}^n$，密钥为 Z_{26} 上的 $n\times n$ 阶可逆方阵 $\boldsymbol{K}=(k_{ij})_{n\times n}$，则

$$加密：密文\ \boldsymbol{c}=\boldsymbol{mK}\ \mathrm{mod}\ 26$$

$$解密：明文\ \boldsymbol{m}=\boldsymbol{cK}^{-1}\ \mathrm{mod}\ 26$$

具体过程如下：

(1) 假设要加密的明文是由 26 个字母组成的。

(2) 将每个字符与 0～25 的一个数字一一对应起来。

(例如，a/A——0，b/B——1，……，z/Z——25)。

(3) 选择一个加密矩阵 $\boldsymbol{A}_{n\times n}$，其中矩阵 $\boldsymbol{A}$ 必须是可逆矩阵，例如

$$\boldsymbol{A}=\begin{bmatrix}7&1&1&5&5\\0&23&18&7&5\\1&10&6&9&2\\16&9&23&21&0\\21&13&7&22&15\end{bmatrix}。$$

(4) 将明文字母分别依照次序每 n 个一组(如果最后一组不足 n 个，就将其补成 n 个)，依照字符与数字的对应关系得到明文矩阵 $\text{ming}_{\text{len}/n\times n}$。

(5) 通过加密矩阵 $\boldsymbol{A}$，利用矩阵乘法得到密文矩阵 $\text{mi}_{\text{len}/n\times n}=\text{ming}_{\text{len}/n\times n}\times\boldsymbol{A}_{n\times n}\ \text{mod}\ 26$。

(6) 将密文矩阵的数字与字符对应起来，得到密文。

(7) 解密时利用加密矩阵的逆矩阵 $\boldsymbol{A}^{-1}$ 和密文，可得到明文。

例如：

随机产生一个 5 阶加密方阵 $\boldsymbol{A}=\begin{bmatrix}7&1&1&5&5\\0&23&18&7&5\\1&10&6&9&2\\16&9&23&21&0\\21&13&7&22&15\end{bmatrix}$

得到方阵 $\boldsymbol{A}$ 的逆矩阵 $\boldsymbol{A}^{-1}=\begin{bmatrix}7&18&4&21&10\\23&7&24&18&1\\12&9&3&20&19\\15&18&23&25&12\\12&4&13&13&9\end{bmatrix}$

加密过程如下。

输入明文：Hill cipher is one of my favorite cipher

分组：Hillc ipher isone ofmyf avori tecip her(aa)

加密得到密文：SKSXAQERQQYDVDGBKNVSMWZATGIAPDOJBIO

解密过程如下。

输入密文：SKSXAQERQQYDVDGBKNVSMWZATGIAPDOJBIO

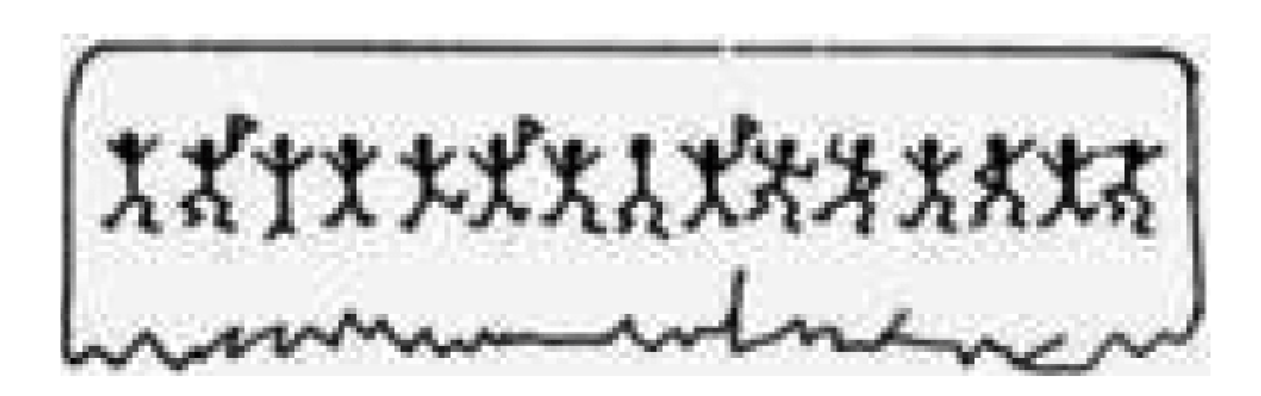
解密得到密文：HILLCIPHERISONEOFMYFAVORITECIPHERAA

2.3.6 多表代替密码——福尔摩斯密码

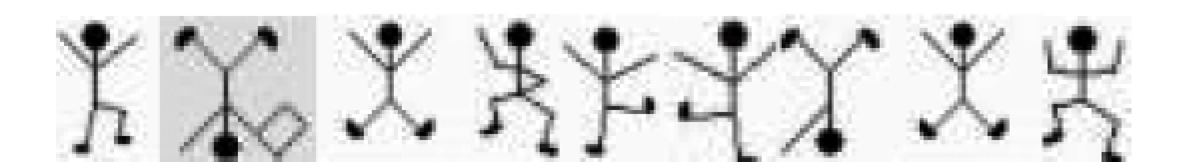
福尔摩斯密码在很多文学作品中都有描述，这里介绍一下亚瑟柯南道尔所写的广为流传的神奇故事——《福尔摩斯探案集之人形密码》，大名鼎鼎的福尔摩斯通过破解加密体制，展示了他非凡的聪明才智。下面是故事中关于密码部分的概要：

希尔顿最近结婚了，他给福尔摩斯发去了一封信，信中有一张纸是他在花园中发现的，

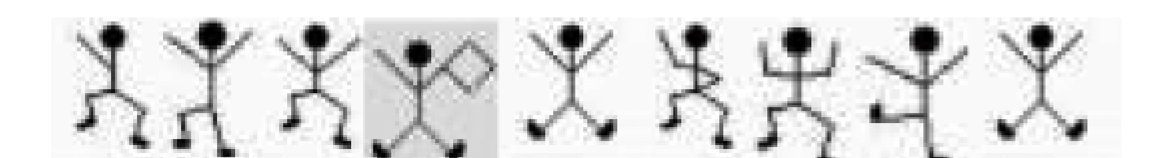
这张纸是用跳舞的棒形小人所写的：



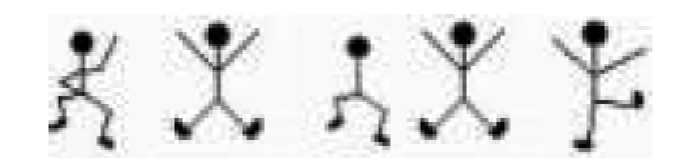
两个星期后，又发现有人用粉笔在他工具间的门上写下了另外一些小人的信息：



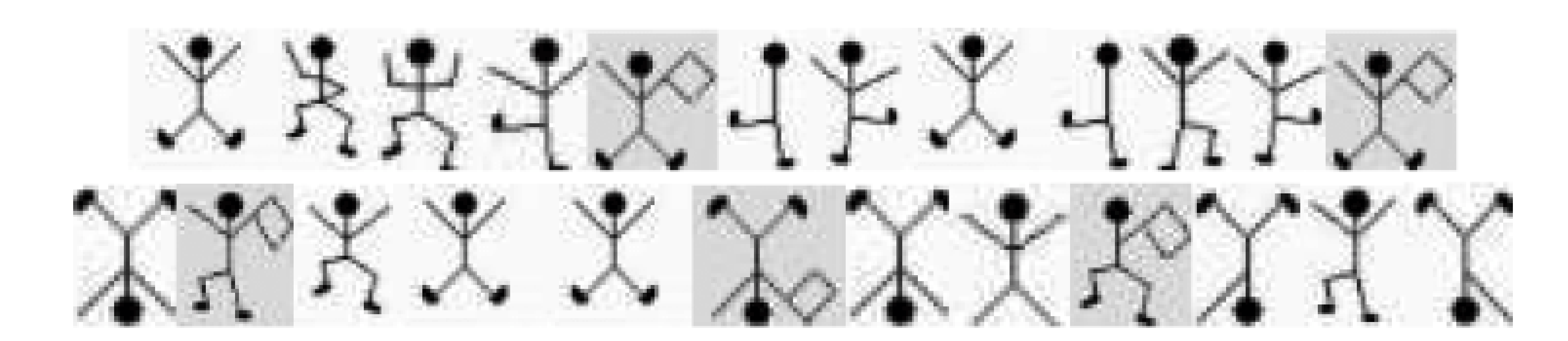
两天以后，又出现了另外的信息：



又过了三天，另一幅小人图出现了：



希尔顿将所有这些小人图拷贝了一份给福尔摩斯，他花了两天时间进行了大量的计算，马上发了一封电报，但两天过去了却没有收到电报的回音，随后收到希尔顿发来的另外一封信：



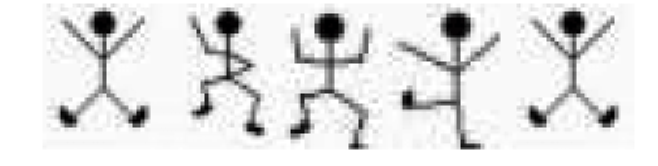
第二天，当福尔摩斯到达希尔顿家时，发现他已被枪杀，他的妻子也受了枪伤并且情况危险。福尔摩斯问了几个问题，并让人给附近农场的艾博送去了一张字条。随后福尔摩斯向警察解释了他是如何解密这些信息的：

首先，他猜测小人手中的旗表示单词结束。

其次他注意到最普通的人是  。

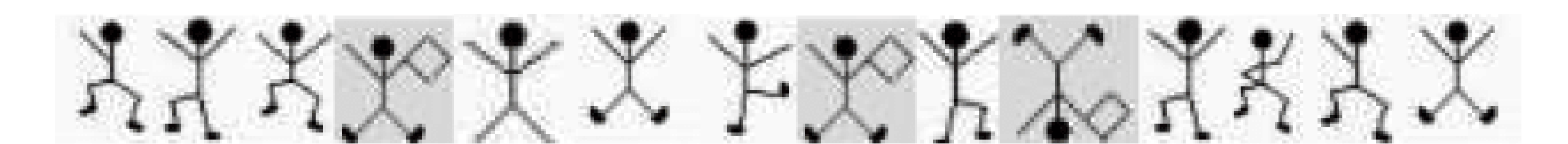
而通常 26 个英文字母中出现频率最高的字母是 E，因此很可能是 E，第 4 个图表示-E-E-信息，很可能是 LEVER，NEVER，SEVER 等含义，但因为很可能这是用一个单词来回复以前的信息，福尔摩斯猜测它是 NEVER。

再次，福尔摩斯观察下面的信息：



有形如 E---E 的形式，它很可能是 ELSIE，第 3 个信息就是---E ELSIE，福尔摩斯想尽了各种组合，最后得出 COME ELSIE 是唯一一种可能的情况。因此第一条信息是-M-ERE--E SL-NE-，福尔摩斯猜测第一个字母是 A，第三个字母是 H，这就表示信息 AM HERE A-ESLANE-，它完整的内容应该是 AM HERE ABE SLANEY。第二条信息就是 A- ELRI-ES，当然，福尔摩斯很正确地猜出这一定表明艾博所在的地方，仅剩余的字母表明的相当完整的短句就是 AT ELRIGES，当最后一封信到来后，解密出的是 ELSIE-RE-ARE TO MEET THY GO-，这样他意识到空缺的字母分别是 P、P、D，于是他开始非常关注，这也就是后来他决定去找希尔顿的原因。

随后，福尔摩斯设局引出凶手艾博。在审讯的过程中，他承认是他开的枪，并说是希尔顿妻子的父亲琼在芝加哥指使的，使用的也是他的枪，但为什么艾博又掉进了福尔摩斯设的圈套呢？福尔摩斯拿出他写的字条：



从字母中已经可以推测出，信息的含义是 COME HERE AT ONCE!

2.4 换位密码

换位就是重新排列消息中的字母，以便打破密文的结构特性，即它交换的不再是字符本身，而是字符被书写的位置。换位分为列换位和周期换位。

2.4.1 列换位

列换位的处理方法是：将明文按照密钥个数排列，并按照密钥在字母表中的顺序变换列的顺序，最后按照列的顺序写出密文。

例 2-7 明文为 cryptography is an applied science，密钥是 creny。

解：根据密钥 creny 中各字母在英文字母表中出现的次序可确定为 14235。将明文按照密钥的长度 5 列逐行列出。

1	4	2	3	5
c	r	y	p	t
o	g	r	a	p
h	y	i	s	a
n	a	p	p	l
i	e	d	s	c
i	e	n	c	e

然后依照密钥决定的次序按列依次读出，因此密文为

cohnii yripdn paspsc rgyaee tpalce。

如果明文不是密钥的整数倍数，那么也需要用其他字母代替，但应事先约定。

2.4.2 周期换位

周期换位的处理方法是：将明文按照密钥个数分组，并按照密钥在字母表中的顺序变换组内字母的顺序，得到密文。

例 2-8 明文为 can you understand，密钥是 fork，求密文。

解：根据密钥 fork 中各字母在英文字母表中出现的次序可确定为 1342。将明文按照密钥的顺序可以得到密文：

密钥顺序	1342	1342	1342	1342
分组	cany	ouun	ders	tand
密文	cyan	onuu	dser	tdan

即密文为 cyanonuudsertdan。

如果明文不是密钥的整数倍，同样需要事先约定用其他字母代替。

在古典密码中，无论是换位密码还是代替密码都是相对简单的密码体制，但其原理与近代密码相似，为近代密码奠定了很好的设计基础。在密码体制设计过程中，一定要遵从 Kerckhoffs 假设。

所谓 Kerckhoffs 假设即一个密码系统的安全强度只能依赖于密钥的保密，而不是加密算法的保密。如果依赖于攻击者不知道算法的内部机理，则注定会失败。

有两点需要注意：

第一，一个加密算法是无条件安全的，如果算法产生的密文不能给出唯一决定相应明文的足够信息。此时无论敌手截获多少密文、花费多少时间，都不能解密密文。

第二，Shannon 指出，仅当密钥至少和明文一样长时，才能达到无条件安全。也就是说除了一次一密方案外，再无其他加密方案是无条件安全的。因此，加密算法只要满足以下两条准则之一即可：

（1）破译密文的代价超过被加密信息的价值。

（2）破译密文所花的时间超过信息的有用期。

满足以上两个准则的加密算法称为计算上安全的。

安全不是一种可以证明的特性，只能说在某些已知攻击下是安全的，对于将来新的攻击是否仍安全就很难断言。

习题

1. 单表代替和多表代替的区别是什么？
2. 周期换位和列换位的区别是什么？
3. 用 Playfair 密码加密明文 playfair cipher，密钥是 PLAYFAIR IS A DIGRAM CIPHER。
4. 用 Vigenere 密码加密明文 We are discovered save yourself，密钥是 deceptive。

第3章 密码学数学基础

在现代密码体制中，构建、分析和攻击这些密码体制都需要用到数学理论，包括数论、有限域、群论等，其中数论是应用最广泛的数学理论。

3.1 素数

3.1.1 整除

定义 3-1 设有整数 a 和 b，且 $b\neq0$。如果存在整数 m，使 $a=mb$，那么就说，a 能被 b 整除，记为 $b|a$，称 b 为 a 的除数。

如 3|15；−15|60，具有以下性质：

(1) 如果 $a|1$，则 $a=\pm1$；

(2) 如果 $a|b$ 且 $b|a$，则 $a=\pm b$；

(3) 对任一非 0 整数 b，$b|0$，$b|b$，$1b$ 均成立；

(4) 如果 $a|b$ 且 $b|c$，则 $a|c$ 成立；

证明：设 $b=k_1\times a \quad c=k_2\times b$，则 $c=k_1\times k_2\times a$。

(5) 如果 $b|g$ 且 $b|h$，则对任意整数 m，n 有 $b|(mg+nh)$。

证明：设 $g=b\times g_1 \quad h=b\times h_1$

则 $mg+nh=mg_1b+nh_1b=(mg_1+nh_1)b$

即 $b|(mg+nh)$

3.1.2 素数

定义 3-2 如果正整数 $P>1$ 只能被 1 和它本身整除，则该数为素数(也叫质数)。

100 以内的素数有 25 个，分别是 2、3、5、7、11、13、17、19、23、29、31、37、41、43、47、53、59、61、67、71、73、79、83、89 和 97。

1000 以内的素数如表 3-1 所示。

表 3-1　1000 以内的素数

2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97	101	103	107	109	113
127	131	137	139	149	151	157	163	167	173
179	181	191	193	197	199	211	223	227	229
233	239	241	251	257	263	269	271	277	281
283	293	307	311	313	317	331	337	347	349
353	359	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463
467	479	487	491	499	503	509	521	523	541
547	557	563	569	571	577	587	593	599	601
607	613	617	619	631	641	643	647	653	659
661	673	677	683	691	701	709	719	727	733
739	743	751	757	761	769	773	787	797	809
811	821	823	827	829	839	853	857	859	863
877	881	883	887	907	911	919	929	937	941
947	953	967	971	977	983	991	997		

定理 3-1　任何大于 1 的整数 a 都可以分解成素数幂之积，且唯一。

$$a = p_1^{a_1} \times p_2^{a_2} \times \cdots\cdots \times p_t^{a_t} \tag{3-1}$$

其中，$p_1 < p_2 < \cdots\cdots < p_t$ 为素数，a_i 为正整数。

例如：$77 = 7^1 \times 11^1$，$504 = 2^3 \times 3^2 \times 7$。

式(3-1)也可以表示为

$$a = \prod_p p^{a_p} (a_p \geqslant 0)。$$

即数 a 是所有素数的乘积。当然，大多的 a_p 都为 0。这样表述的优点在于，两个数的乘法等于对应素数指数的加法。

例如：$6 = 2 \times 3$，$18 = 2 \times 3^2$，则 $6 \times 18 = 2^2 \times 3^3$。

对于 $a|b$，它们的素数因子关系为

$$a|b \rightarrow a_p < b_p（对每一项的素数都如此）$$

素数在密码学中具有重要的作用，尤其是在非对称密码体制中，经常用到很大的素数。2008 年 9 月，德国人发现的素数为 1300 万位的整数，用 5 号铅字将其印刷，它的长度将达到 30 英里。

3.1.3　最大公约数

定义 3-3　a 和 b 的最大公约数是能够同时整除 a 和 b 的最大正整数，记为

$$\gcd(a,b)。$$

例如：$\gcd(6,4) = 2$；$\gcd(3,7) = 1$。

如果 $\gcd(a,b) = 1$，那么就说 a 和 b 是互素的。

下面分析如何求解两个数的最大公约数。

（1）对于不是很大的数，利用素数来求解。

例如：$1728=2^6\times3^3$　　$135=3^3\times5$，则 $\gcd(1728,135)=3^3=27$。

（2）对于很大的数，将其分解为素数乘积的形式比较困难，可以用欧几里得算法求解。

$$\gcd(a,b)=\gcd(b,a \bmod b) \tag{3-2}$$

例如：gcd(12 345,1111)＝gcd(1111,124)＝ gcd(124,5)＝gcd(5,4)＝ gcd(4,1)＝gcd(1,0)

到最后的 $a \bmod b=0$ 为止，则最后的 b 为所求，所以 12 345 与 1111 的最大公约数为 1，即它们是互素的。

3.2　模运算

定义 3-4　设整数 a、b 及 $n\neq0$，若 $a-b=kn$（k 为任一整数），则称 a 在 mod n 下与 b 同余，记为 $a\equiv b \bmod n$。

例如：$11 \bmod 7=4$；$4 \bmod 7=4$；则 $11\equiv4 \bmod 7$。

模运算有以下性质：

（1）$a\equiv a \bmod n$。

（2）若 $a\equiv b \bmod n$，则 $b\equiv a \bmod n$。

（3）若 $a\equiv b \bmod n$ 且 $b\equiv c \bmod n$ ，则 $a\equiv c \bmod n$。

（4）若 $a\equiv b \bmod n$ 且 $c\equiv d \bmod n$，则 $a+c\equiv(b+d) \bmod n$，$a-c\equiv(b-d) \bmod n$，$ac\equiv(bd) \bmod n$。

（5）$(a+b) \bmod n=((a \bmod n)+(b \bmod n)) \bmod n$，

$(a-b) \bmod n=((a \bmod n)-(b \bmod n)) \bmod n$，

$(a\times b) \bmod n=((a \bmod n)\times(b \bmod n)) \bmod n$。

（6）若 $ac\equiv(bd) \bmod n$，$c\equiv d \bmod n$ 且 $\gcd(c,n)=1$，则 $a\equiv b \bmod n$。

（7）若 $\gcd(a,n)=1$，则存在唯一整数 b，$0<b<n$ 且 $\gcd(b,n)=1$，使得 $ab\equiv1 \bmod n$。此时 a 称为 b 在 mod n 下的反元素；b 称为 a 在 mod n 下的反元素。

（8）若 $ax\equiv1 \bmod n$ 有解，则 $\gcd(a,n)=1$。

例 3-1　计算 $11^7 \bmod 13$。

解：利用 $(a\times b) \bmod n=((a \bmod n)\times(b \bmod n)) \bmod n$，得

$$\begin{aligned}11^7 \bmod 13&=(11\times11^2\times11^4) \bmod 13\\&=(11\times4\times3) \bmod 13\\&=132 \bmod 13\\&=2\end{aligned}$$

例 3-2　证明 $5^{60}-1$ 是 56 的倍数。

该问题等价于 $5^{60} \bmod 56=1$。

证明：$5^3 \bmod 56=13$

$5^6 \bmod 56=13^2 \bmod 56=1$

$5^{60} \bmod 56=1^{10} \bmod 56=1$

即 $5^{60}-1$ 是 56 的倍数。

3.3 模逆元

在密码学中,经常用到模逆元。

定义 3-5 模逆元是指寻找一个最小正整数 x,使 $ax \equiv 1 \bmod n$,这里 a 和 n 皆为正整数,且 a 与 n 互素,x 称为 a 的模 n 逆元,记为 $x=a^{-1} \bmod n$。

如果 a 与 n 不互素,那么不存在 x,使 $ax \equiv 1 \bmod n$。

模逆元的计算可以通过扩展欧几里得算法实现。

扩展欧几里得算法可以描述为

(1) $(X_1, X_2, X_3) \leftarrow (1, 0, n)$; $(Y_1, Y_2, Y_3) \leftarrow (0, 1, a)$。

(2) 如果 $Y_3=0$,返回 $X_3=\gcd(a,n)$; 无逆元。

(3) 如果 $Y_3=1$,返回 $Y_3=\gcd(a,n)$; $Y_2=a^{-1} \bmod n$。

(4) $Q=\lfloor X_3/Y_3 \rfloor$(即除数,并往下取整)。

(5) $(T_1, T_2, T_3) \leftarrow (X_1-QY_1, X_2-QY_2, X_3-QY_3)$。

(6) $(X_1, X_2, X_3) \leftarrow (Y_1, Y_2, Y_3)$。

(7) $(Y_1, Y_2, Y_3) \leftarrow (T_1, T_2, T_3)$。

(8) 返回第(2)步。

如果有逆元,Y_2 为逆元,$Y_3=\gcd(a,n)$是 a 和 n 的最大公约数。

例 3-3 用扩展欧几里得算法求 gcd(7,26)和 $7^{-1} \bmod 26$。

解:

Q	X_1	X_2	X_3	Y_1	Y_2	Y_3
	1	0	26	0	1	7
3	0	1	7	1	−3	5
1	1	−3	5	−1	4	2
2	−1	4	2	3	−11	1

所以 $\gcd(7,26)=1$,$7^{-1} \bmod 26 = -11 \bmod 26 = 15$。

3.4 费马欧拉定理

3.4.1 费马定理

定理 3-2(费马定理 1) 如果 p 是素数,且 p 不能被 a 整除,那么 $a^{p-1} \equiv 1 \bmod p$。例如:$a=5$,$p=11$

$$\begin{aligned} a^{p-1} \bmod p &= 5^{10} \bmod 11 = (5^3 \times 5^3 \times 5^3 \times 5) \bmod 11 \\ &= (64 \times 5) \bmod 11 = 45 \bmod 11 = 1 \end{aligned}$$

定理 3-3(费马定理 2)　如果 p 是素数,a 是正整数,且 $\gcd(a,p)=1$,那么 $a^p \equiv a \bmod p$。例如:$a=2,p=5$

$$a^p \bmod p = 2^5 \bmod 5 = 32 \bmod 5 = 2$$

3.4.2　欧拉定理

定义 3-6　当 $m>1$ 时,欧拉函数 $\varphi(m)$表示比 m 小,且与 m 互素的正整数的个数。例如:$m=12$,比 12 小且与 12 互素的正整数为 1、5、7、11,所以

$$\varphi(12) = 4$$

欧拉函数具有以下性质:

(1) 当 m 是素数时,$\varphi(m)=m-1$,即比 m 小的所有正整数,如 $\varphi(11)=10$。

(2) 当 $m=pq$,且 p、q($p\neq q$)均为素数时,$\varphi(m)=\varphi(p)\varphi(q)=(p-1)(q-1)$。

证明:$m=pq$,比 m 小的正整数的集合 $Z=\{1,2,\cdots\cdots,pq-1\}$。

在集合 Z 中,与 m 不互素的数为 p 的倍数和 q 的倍数。

p 的倍数的集合为$\{p,2p,\cdots\cdots,(q-1)p\}$,共$(q-1)$个数。

q 的倍数的集合为$\{q,2q,\cdots\cdots,(p-1)q\}$,共$(p-1)$个数。

所以,$\varphi(m)=(pq-1)-(q-1)-(p-1)=(p-1)\times(q-1)=\varphi(p)\varphi(q)$。

例如:$m=15=3\times5$, $\varphi(m)=2\times4=8$

比 15 小且与 15 互素的正整数为 1、2、4、7、8、11、13、14,所以 $\varphi(15)=8$。

(3) 当 $m=p^2$,且 p 为素数时,$\varphi(m)=p(p-1)$。

证明:$m=p^2$,比 m 小的正整数的集合 $Z=\{1,2,\cdots\cdots,p^2-1\}$。

在集合 Z 中,与 m 不互素的数为 p 的倍数。

p 的倍数的集合为$\{p,2p,\cdots\cdots,(p-1)p\}$,共$(p-1)$个数,

所以,$\varphi(m)=(p^2-1) - (p-1)=p(p-1)$。

例如:$m=9=3^2$, $\varphi(m)=3\times2=6$

比 9 小且与 9 互素的正整数为 1、2、4、5、7、8,所以 $\varphi(9)=6$。

当计算一个数的欧拉函数 $\varphi(m)$时,可以采用以下两个公式进行计算。

(1) 若一个数 m 可以写成 $m=p_1^{e_1}\times p_2^{e_2}\times\cdots\cdots\times p_t^{e_t}$($p_i$ 为素数),则

$$\varphi(m) = \prod_{i=1}^{t} p_i^{e_i-1}(p_i - 1) \tag{3-3}$$

例如:$m=120=2^3\times3\times5$

$\varphi(m)=2^2\times(2-1)\times(3-1)\times(5-1)=32$

(2) 对任一正整数 m,若其可写成 $p_1^{e_1}\times p_2^{e_2}\times\cdots\cdots\times p_t^{e_t}$,则

$$\varphi(m) = m \times \prod_{p_i}\left(1-\frac{1}{p_i}\right) \tag{3-4}$$

例如:$m=120$

$\varphi(m)=120\times\left(1-\frac{1}{2}\right)\left(1-\frac{1}{3}\right)\left(1-\frac{1}{5}\right)=32$

定理 3-4(欧拉定理)　对于任何互素的两个整数 a 和 n,有 $a^{\varphi(n)}\equiv1 \bmod n$。

欧拉定理具有以下性质:

（1）当 n 为素数时，欧拉定理相当于费马定理；

（2）$a^{\varphi(n)+1} \equiv a \bmod n$；

（3）$(a^{\varphi(n)})^k \equiv 1 \bmod n$。

例如：$a=7, n=10, \varphi(10)=4, 7^4 \bmod 10=1$。

例 3-4 求 7^{803} 的后三位数字。

解：原题等价于 $7^{803} \bmod 1000$。

$1000=2^3 \times 5^3$，所以 $\varphi(1000)=1000\times\left(1-\frac{1}{2}\right)\left(1-\frac{1}{5}\right)=400$

由于 7 与 1000 互素，根据欧拉定理得

$$7^{\varphi(1000)} \bmod 1000 = 1$$

$$7^{400} \bmod 1000 = 1$$

所以 $7^{803} \bmod 1000=(7^3 \times 7^{400} \times 7^{400}) \bmod 1000=343$，即 7^{803} 的后三位数字为 343。

3.4.3 本原元

定义 3-7 对于任何互素的两个整数 a 和 n，在方程 $a^m=1 \bmod n$ 中，至少有一个正整数 m 满足这一方程（因为 $\varphi(n)$ 是其中的一个解），那么，最小的正整数解 m 为模 n 下 a 的阶。如果 a 的阶 $m=\varphi(n)$，称 a 为 n 的本原元。

例如：$a=7$，$n=10, \varphi(10)=4$

在方程 $7^m=1 \bmod 10$ 中，$7^1=7 \bmod 10, 7^2=9 \bmod 10, 7^3=3 \bmod 10, 7^4=1 \bmod 10$，而 $\varphi(10)=4$，所以 7 为 10 的本原元。

例如：$a=7$，$n=19, \varphi(19)=18$

在方程 $7^m=1 \bmod 19$ 中，$7^1=7 \bmod 19, 7^2=11 \bmod 19, 7^3=1 \bmod 19$，而 $\varphi(19)=18$，所以 7 不是 19 的本原元。

这里需要注意两点：

（1）一个数的本原元不唯一；

（2）有些数没有本原元。

例如：$a=2$，$n=19, \varphi(19)=18$

经过计算 2 是 19 的本原元。同理，3、10、13、14、15 都是 19 的本原元。

例如：$a=3, n=8, \varphi(8)=4$

经过计算 3 不是 8 的本原元。同理 5 和 7 也不是 8 的本原元，所以 8 没有本原元。

概括地说，只有 2、4、p^a、$2p^a$ 有本原元，其中 a 为正整数，p 为奇素数。

3.5 中国余数定理

孙子算经：今有物不知其数，三三数之剩二，五五数之剩三，七七数之剩二，问物几何？

其含义其实是求正整数解 x 满足：

$$\begin{cases} x = 2 \bmod 3 \\ x = 3 \bmod 5 \\ x = 2 \bmod 7 \end{cases}$$

定理 3-5 令 $n_1,n_2,\cdots,n_t$ 为两两互质的正整数，$N=\prod_{i=1}^{t} n_i$，则 $x=a_1 \bmod n_1=a_2 \bmod n_2=\cdots=a_t \bmod n_t$ 在$[0,N-1]$中有唯一解，称为中国余数定理。

下面来分析如何求中国余数定理的解。

令 $N_i=\frac{N}{n_i}$，则 $x=\sum_{i=1}^{t} N_i \times y_i \times a_i \bmod N$，其中

$N_i \times y_i=1(\bmod n_i)$，亦即 $y_i=N_i^{-1} \bmod n_i$。

孙子算经之例子解法如下：

$$N = 3 \times 5 \times 7 = 105$$
$$N_1 = 5 \times 7 = 35, N_2 = 3 \times 7 = 21, N_3 = 3 \times 5 = 15$$
$$35 \times y_1 = 1 \bmod 3 \Rightarrow y_1 = 2$$
$$21 \times y_2 = 1 \bmod 5 \Rightarrow y_2 = 1$$
$$15 \times y_3 = 1 \bmod 7 \Rightarrow y_3 = 1$$

所以，$x=35\times2\times2+21\times1\times3+15\times1\times2=23 \bmod 105$。

中国余数定理的计算可以总结为

给定 a_1,a_2,n_1,n_2 且 $n_1<n_2$，$\gcd(n_1,n_2)=1$，求 x，使得 $0\leqslant x<n_1 \cdot n_2$ 并满足 $x=a_1 \bmod n_1=a_2 \bmod n_2$。

解法如下：

首先，求出 u 满足 $u \cdot n_2=1 \bmod n_1$。

(1) 若 $a_1\geqslant(a_2 \bmod n_2)$，则

$$x = (((a_1 - (a_2 \bmod n_1)) \cdot u) \bmod n_1) \cdot n_2 + a_2$$

(2) 若 $a_1<(a_2 \bmod n_2)$，则

$$x = (((a_1 + n_1 - (a_2 \bmod n_1)) \cdot u) \bmod n_1) \cdot n_2 + a_2$$

在非对称密码算法 RSA 中，利用中国余数定理可以使得解密速度约为原来的四倍。应用中国余数定理可解决安全广播系统、密钥确认、存取控制等问题。

3.6 单向函数与单向暗门函数

一个单向函数 $f:X\rightarrow Y$，应满足下列条件：

(1) 对任一 $x\in Y$，可以很容易算出 $y=f(x)$。

(2) 给定任一 $y\in Y$，算出 x 满足 $y=f(x)$在计算上不可行。

一个单向暗门函数 $f:X\rightarrow Y$，应满足下列条件：

(1) 对任一 $x\in X$，可以很容易算出 $y=f(x)$。

(2) 给定任一 $y\in Y$，算出 $x=f^{-1}(y)$在计算上不可行；若知道某一个额外的秘密参数(称为暗门)，则可以很容易算出 $x=f^{-1}(y)$。

单向函数的应用如将某一个秘密值转换成一个公开值，而任何人无法从公开值中求得该秘密值。例如，将密钥转换成一个公开值存放于一个公开目录中，握有真正密钥的人可以将其密钥先经过单向函数转换后，再与存放于公开目录的公开值加以比对，而达到验证身分

的目的。

单向暗门函数的应用如将某一个秘密值转换成一个公开值后，借由暗门可以将该公开值反解成原来的秘密值。例如，加解密运算的暗门为密钥。

例如，在多项式中，令 $y=f(x)=(a_0+a_1x+\cdots+a_{n-1}x^{n-1}+x^n)\ \mathrm{mod}\ p$。若给定 a_0，a_1，…，a_n、p 和 x，很容易算出 $y=f(x)$。但若给定 a_0，a_1，…，a_n、p 和 y，欲求出 $f(x)$的根 x，则至少需要 $n^2(\log_2 p)^2$ 个乘法。当 n 与 p 很大时，求出 $f(x)$的根 x 是相当困难的。

再如，在离散对数中，令 p 为质数且 $p-1$ 含有一个大质因子 q。给定一整数 g、$1<g<p-1$ 与 x，计算 $y=g^x\ \mathrm{mod}\ p$ 最多仅需$\lfloor \log_2 x \rfloor+w(x)-1$ 个乘法，其中 $w(x)$表示二进制表示法 x 内 1 的个数。反之，给定一整数 g 与 y，求出 x 满足 $y=g^x\ \mathrm{mod}\ p$ 需要 $\exp\{(\ln p \ln(\ln p))^{1/2}\}$次运算。

习题

1. 一个数的本原根是什么？
2. 对两个连续的整数 n 和 $n+1$，为什么 $\gcd(n,n+1)=1$？
3. 利用 Fermat 定理计算 $3^{201}\ \mathrm{mod}\ 11$。
4. 找出 25 的所有本原根。
5. 用扩展欧几里得算法求 $\gcd(7,31)$和 $7^{-1}\ \mathrm{mod}\ 31$。
6. 利用中国剩余定理求解下式。

$$\begin{cases} x\equiv 2\ \mathrm{mod}\ 3\\ x\equiv 1\ \mathrm{mod}\ 5\\ x\equiv 1\ \mathrm{mod}\ 7\end{cases}$$

第4章 分组加密技术

4.1 分组密码

4.1.1 分组密码概述

分组密码是将明文消息编码得到的数字序列划分成长为 n 的组，各组分别在密钥控制下变换成等长的输出数字序列。

在相同的密钥下，分组密码对长为 n 的输入明文组所实施的变换是等同的，所以只需研究对任一组明文数字的变换规则。这种密码实质上是字长为 n 的数字序列的代换密码。分组密码原理图如图 4-1 所示。将明文转化为二进制后，对其分组，每组采用相同的加密体制，并分别得到密文，再转化成字符。

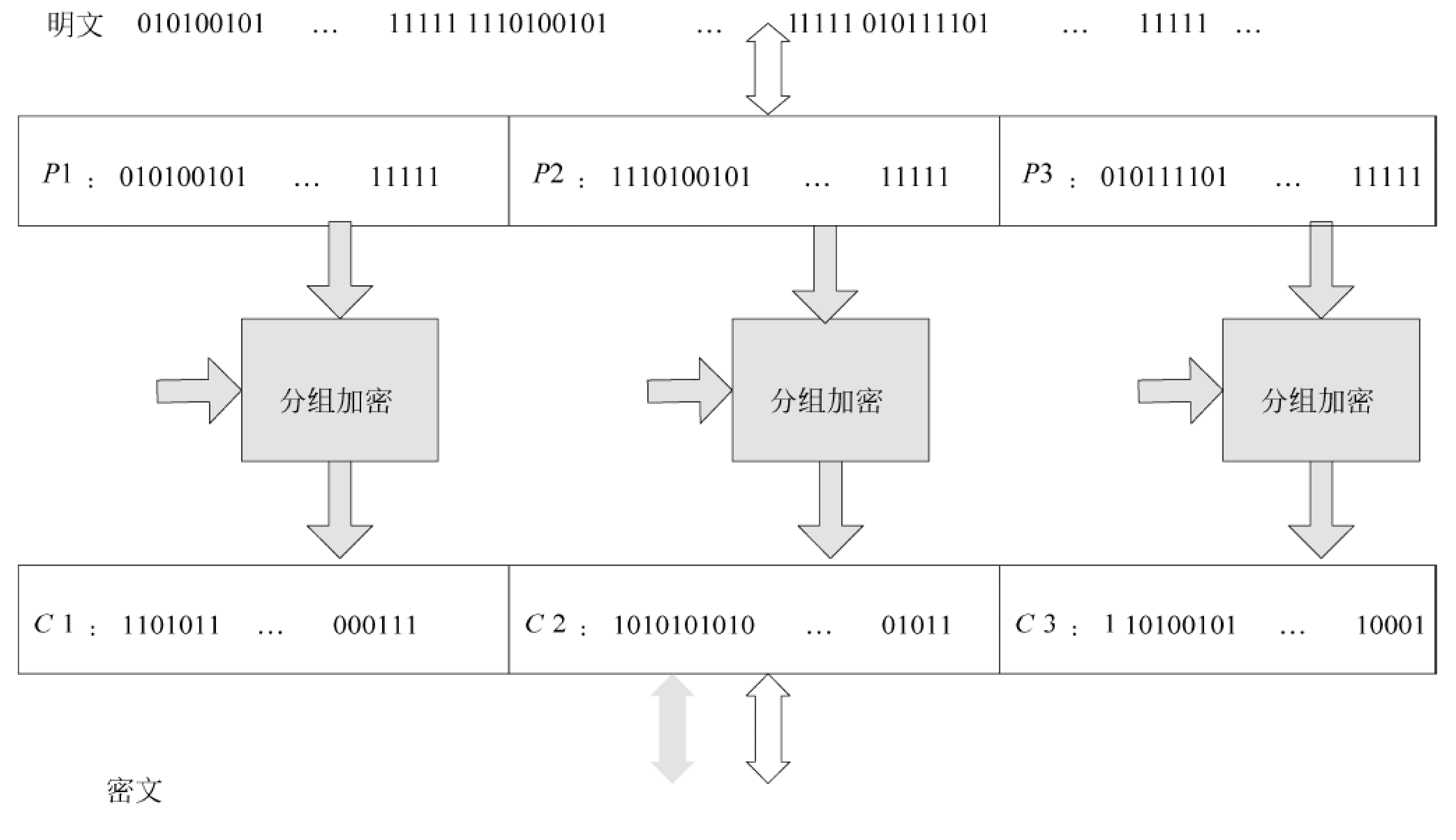


图 4-1 分组密码原理图

为了保证加密算法的安全性，对密码算法有以下要求：

(1) 分组长度足够大。分组长度 n 要足够大，使分组代换字母表中的元素个数 2^n 足够大，防止明文穷举攻击法奏效。DES、IDEA 和 FEAL 等分组密码都采用 $n=64$，使穷举攻击很难实现。

(2) 密钥量足够大。密钥量要足够大，尽可能消除弱密钥并使所有密钥同等地好，以防止密钥穷举攻击奏效。但密钥又不能过长，以便于密钥的管理。

(3) 由密钥确定置换的算法要足够复杂。充分实现明文与密钥的扩散和混淆，没有简单的关系可循，能抗击各种已知的攻击，使对手破译时除了用穷举法外，无其他捷径可循。

(4) 加密和解密运算简单，易于软件和硬件高速实现。在以软件实现时，应选用简单的运算，使作用于子段上的密码运算易于标准处理器的基本运算，如加、乘、移位等；为了便于硬件实现，加密和解密过程之间的差别应仅在于由秘密密钥所生成的密钥表不同而已。这样，加密和解密就可用同一器件实现。

4.1.2 分组密码设计思想

扩散和混淆是由 Shannon 提出的设计密码系统的两个基本方法，目的是抗击敌手对密码系统的统计分析。扩散是将明文的统计特性散布到密文中去，实现方式是使得明文的每一位影响密文中多位的值，等价于密文中的每一位均受明文中多位的影响；混淆是使密文和密钥之间的统计关系变得尽可能复杂，以使敌手无法得到密钥。

Horst Feistel(IBM Labs)在 20 世纪 70 年代初，设计了 Feistel 网络结构，提出利用乘积密码可获得简单的代换密码，乘积密码是指顺序地执行两个或多个基本密码系统，使得最后结果的密码强度高于每个基本密码系统产生的结果。图 4-2 是 Feistel 的网络示意图。

Feistel 网络中每轮结构都相同，每轮中右半数据被作用于轮函数 F 后，再与左半数据进行异或运算，这一过程就是上面介绍的代换。每轮的轮函数结构都相同，但以不同的子密钥 K 作为参数。代换过程完成后，再交换左、右两半数据，这一过程称为置换。这种结构是 Shannon 提出的代换-置换网络的特有形式。

Feistel 网络的实现与以下参数和特性有关：

(1) 分组大小。分组越大则安全性越高，但加密速度就越慢。分组密码设计中最为普遍使用的分组大小是 64 比特。

(2) 密钥大小。密钥越长则安全性越高，但加密速度就越慢。现在通常使用 128 比特的密钥长度。

(3) 轮数。单轮结构远不足以保证安全性，多轮结构可提供足够的安全性。典型地，轮数取为 16。

(4) 子密钥产生算法。该算法的复杂性越大，则密码分析的困难性就越大。

(5) 轮函数。轮函数的复杂性越大，密码分析的困难性也越大。

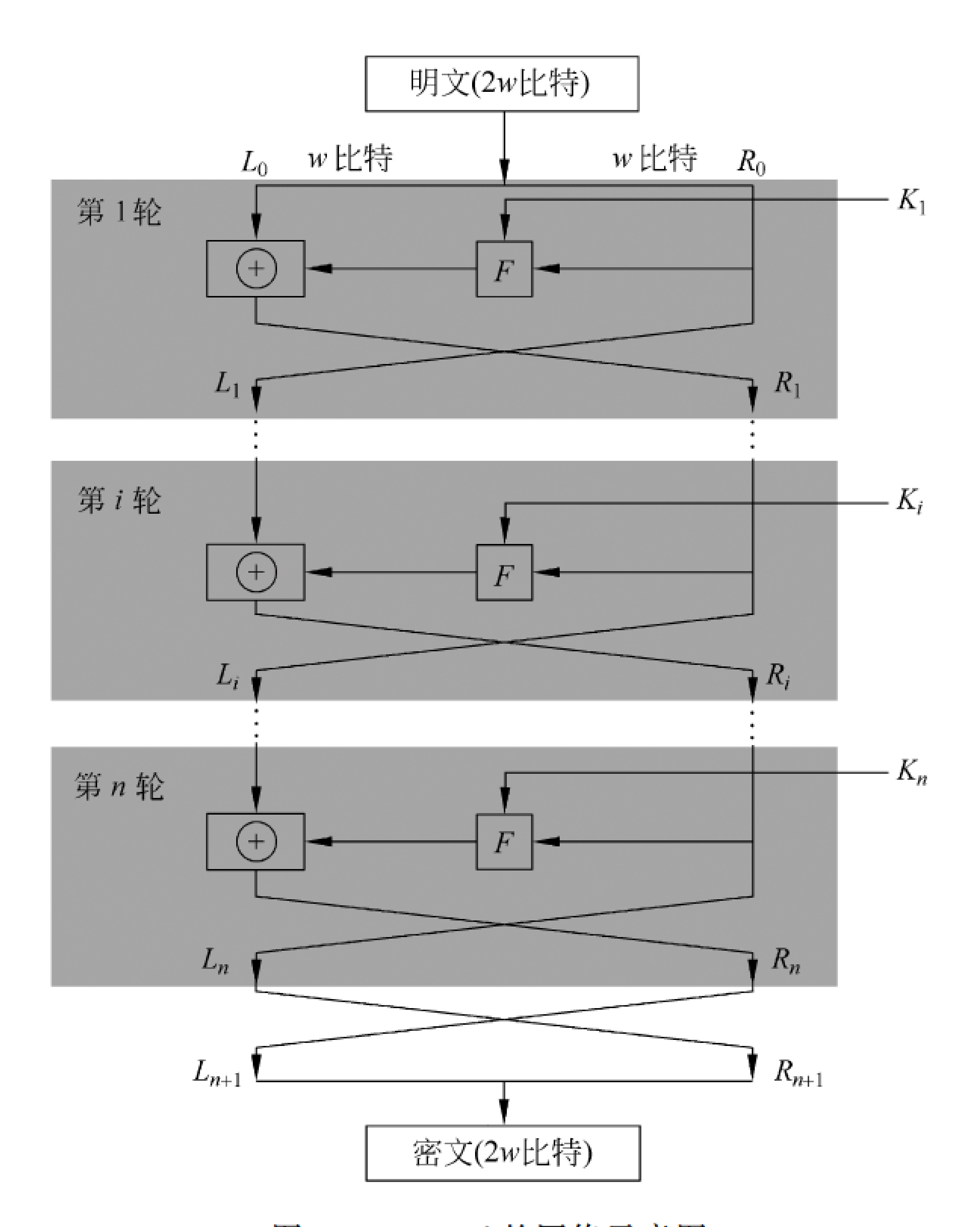


图 4-2 Feistel 的网络示意图

4.2 S-DES

数据加密标准(Data Encryption Standard,DES)是迄今为止世界上最为广泛使用和流行的一种分组密码算法,它的分组长度为 64 比特,密钥长度为 56 比特,16 轮迭代。它是由美国 IBM 公司研制的,基于 Feistel 网络结构。DES 在 1975 年 3 月 17 日首次被公布在联邦记录中,经过大量的公开讨论后,DES 于 1977 年 1 月 15 日被正式批准并作为美国联邦信息处理标准,即 FIPS-46,同年 7 月 15 日开始生效,作为数据加密标准。

为了更好地理解 DES 算法,美国圣塔克拉拉大学的 Edward Schaefer 教授于 1996 年开发了 Simplified DES 方案,简称 S-DES 方案。它是一个供教学而非安全的加密算法,它与 DES 的特性和结构类似,但参数小,明文分组为 8 位,主密钥分组为 10 位,采用两轮迭代。

4.2.1 S-DES 加密原理

S-DES 的加密原理如图 4-3 所示。

S-DES 的具体实现步骤如下。

(1) 初始置换 IP:将 8 位的明文按照置换顺序(26314857)进行位置变化,置换后分为左 4 位 L_0 和右 4 位 R_0。

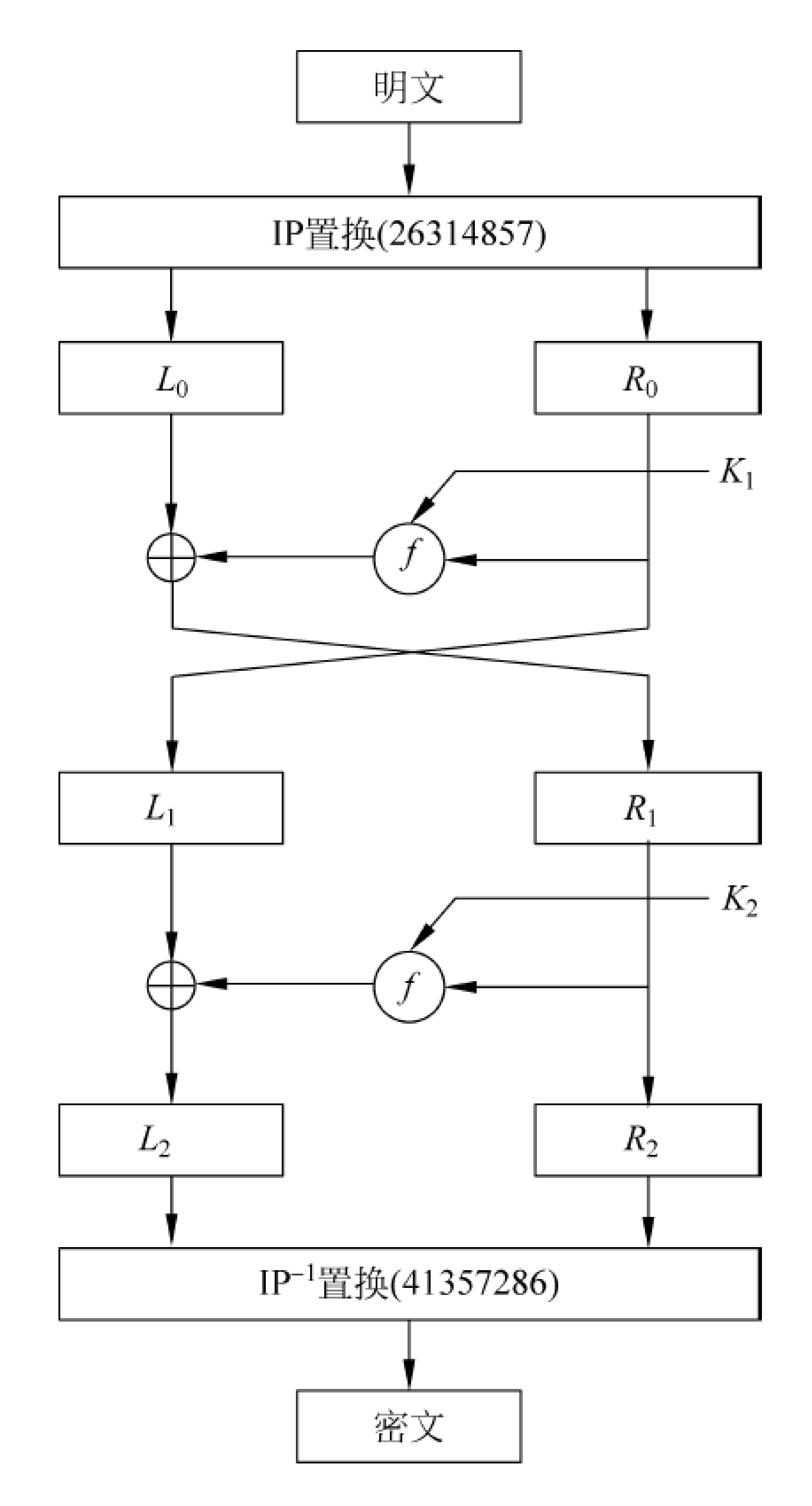


图 4-3 S-DES 的加密原理图

(2) 第 1 轮运算,R_0一方面直接输出作为下一轮的 L_1,另一方面作为 f 函数的输入与 8 位的子密钥 K_1参与函数运算,运算结果与 L_0异或,结果作为下一轮的 R_1。

(3) 第 2 轮运算,R_1一方面直接输出作为下一轮的 R_2,另一方面作为 f 函数的输入与 8 位的子密钥 K_2参与函数运算,运算结果与 L_1异或,结果作为下一轮的 L_2。

(4) 逆置换 IP^{-1}。在 S-DES 的整个加密过程中,包含两个重要的组成部分,一个是子密码生成过程,一个是 f 函数结构。

4.2.2 S-DES 的子密码生成过程

在图 4-3 中的子密钥 K_1和 K_2,是由 10 位主密钥产生的。S-DES 的子密钥生成过程如图 4-4 所示。

S-DES 子密钥的生成过程是将一个 10 比特的主密钥 K 生成两个 8 比特的子密钥,其步骤如下。

(1) 主密钥 K 进行 $P10$ 置换(3、5、2、7、4、10、1、9、8、6)。

(2) 分成左 5 位和右 5 位,再分别进行 LS—1 操作(左循环 1 位),其结果一方面作为下一轮的初始值,一方面进行 $P8$ 置换(6、3、7、4、8、5、10、9),得到 K_1($P8$ 相当于将 10 位截取成 8 位)。

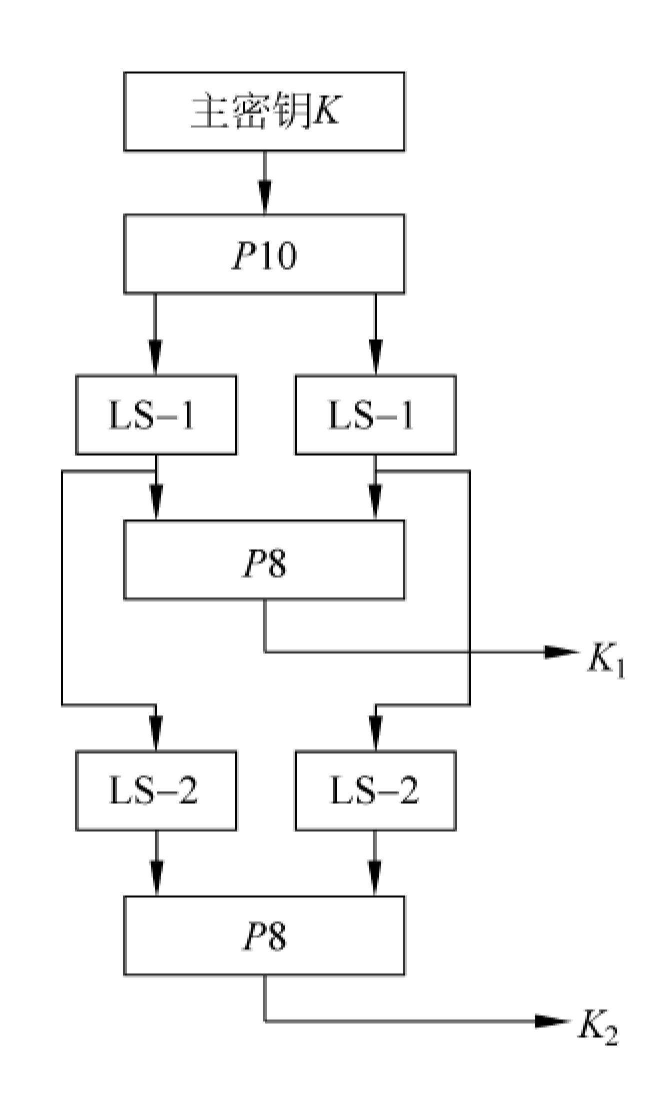


图 4-4 S-DES 的子密码生成过程

(3) 再分别左循环 2 位，经过 $P8$ 置换，得到 K_2。

例 4-1 主密钥 K=10100 00010，求子密钥 K_1 和 K_2。

解：(1) 主密钥经过 $P10$ 置换得到：10000 01100。

(2) LS—1 操作：00001 11000。

(3) $P8$ 置换产生 K_1：10100100。

(4) LS—2 操作：00100 00011。

(5) $P8$ 置换产生 K_2：0100 0011。

所以，K_1：10100100，K_2：0100 0011。

4.2.3 S-DES 的 f 函数结构

S-DES 的 f 函数结构如图 4-5 所示。

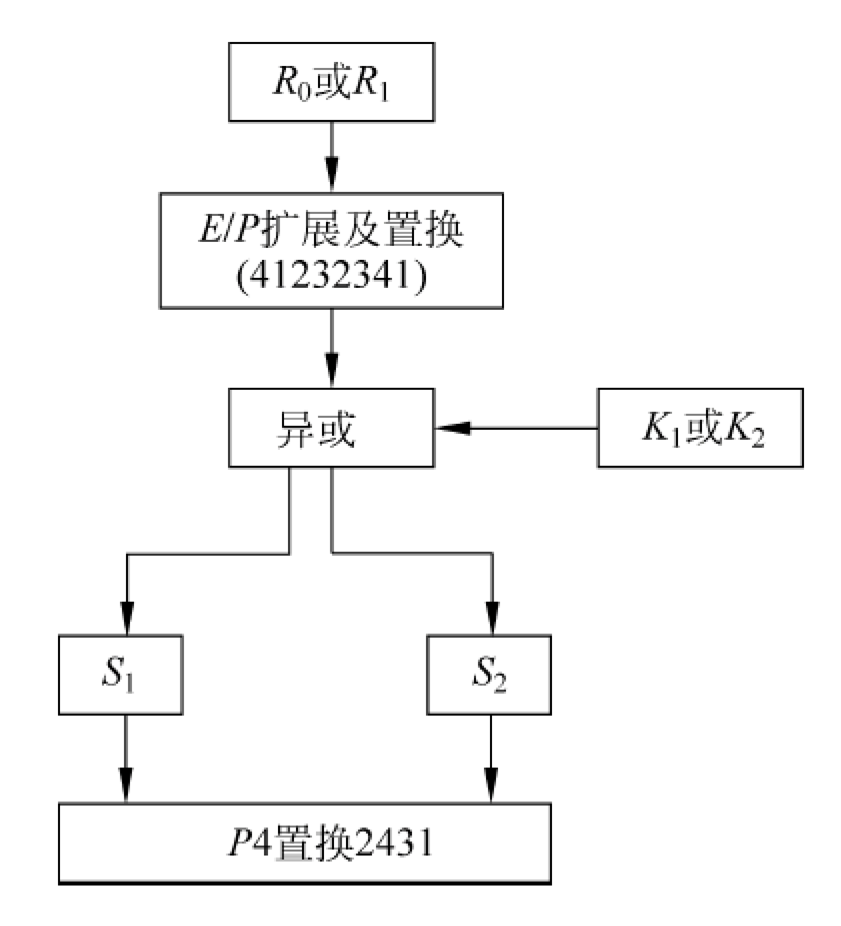


图 4-5 S-DES 的 f 函数结构

S-DES 的 f 函数结构的具体实现步骤如下。

（1）E/P 扩展及置换：将 4 位 R_0 或 R_1 扩展为 8 位。

（2）扩展后的 8 位与密钥 K_1 或 K_2 异或，输出 8 位。

（3）左边 4 位作为 S_1 盒输入，右边作为 S_2 盒输入。

（4）在 S_1 和 S_2 中，第 1 位与第 4 位结合形成 2 位代表 S 盒的行号，第 2 位与第 3 位结合形成 2 位代表 S 盒的列号，得到 S 盒的输出。S-DES 的 S 盒如表 4-1 所示。

（5）进行 $P4$ 置换，得到 f 函数的输出。

表 4-1　S-DES 的 S 盒

S_1	00	01	10	11	S_2	00	01	10	11
00	01	00	11	10	00	00	01	10	11
01	11	10	01	00	01	10	00	01	11
10	00	10	01	11	10	11	10	01	00
11	11	01	00	10	11	10	01	00	11

例 4-2　设 $R_0=0101, K_1=10100100$，求 f 函数的输出。

解：（1）E/P 扩展及置换：10101010。

（2）与 $K1$ 异或：00001110。

（3）S_1 输入＝0000　S_2 输入＝1110

S_1 行号＝00　S_1 列号＝00　S_1 输出＝01

S_2 行号＝10　S_2 列号＝11　S_2 输出＝00

（4）$P4$ 置换：f 输出＝1000。

例 4-3　设主密钥 K＝10100 00010，用 S-DES 加密明文字母 C。

解：（1）计算字密钥：由例 4-1 的结果得 K_1：10100100，K_2：0100 0011。

（2）字母 C 的 ASCII 为 67，对应的二进制为 0100 0011。

（3）IP 置换：$L_0=1000, R_0=0101$。

（4）第 1 轮：$f(R_0, K_1)=1000$

$$L_0 \oplus f = 1000 \oplus 1000 = 0000$$

（5）$L_1 = R_0 = 0101$；$R_1 = 0000$。

（6）第 2 轮：$f(R_1, K_2)=1001$

$$L_1 \oplus f = 0101 \oplus 1001 = 1100$$

（7）$L_2 = 1100$；$R_2 = R_1 = 0000$。

（8）IP^{-1} 置换：0100 0100（ASCII 为 68，对应字母 D）。

4.3　美国数据加密标准

DES 是在 20 世纪 70 年代由美国 IBM 公司发展起来的，且被美国国家标准局公布为数据加密标准的一种区块加密法。所谓的区块加密法是对一定大小的明文或密文来做加密或解密动作，而在 DES 加密系统中，每次加密或解密的区块大小均为 64 位，因此 DES 没有密文扩充的问题。

就一般数据而言，无论明文或密文，其数据大小都大于 64 位。这时只要将明密文中每 64 位当成一个区块加以切割，再对每一个区块做加密或解密即可。另外，DES 所用的加密或解密密钥也是 64 位大小，但其中有 8 个位是用来做错误更正的，所以 64 位中真正是有密钥效用的只有 56 位。

DES 的设计准则如下。

(1) 随机性：输出与输入无规律。

(2) 雪崩效应：改变输入中的 1 位，会导致输出一半以上的位被改变。

(3) 完全性：每个输出位是所有输入位的一个复杂函数，而不是某个或某些位的函数。

(4) 非线性：加密函数和密钥之间是非线性的。

(5) 相关免疫性：明文和密文不存在相关性。

4.3.1 DES 加密原理

DES 加密算法的原理框图如图 4-6 所示。

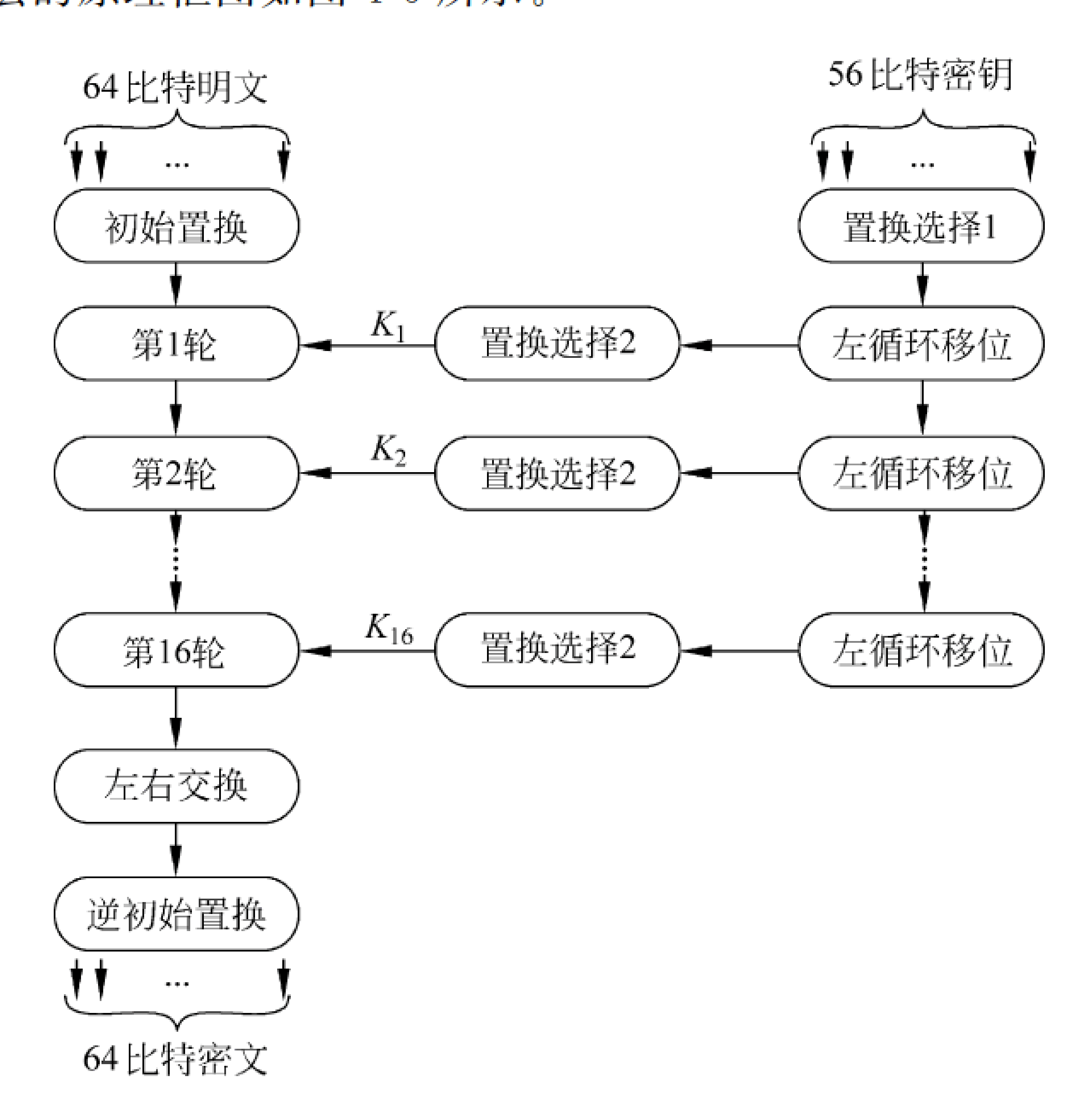


图 4-6 DES 加密算法原理框图

具体的加密步骤如下。

(1) 令明文 $M=t_1t_2\cdots t_{64}$。将明文进行初始置换，进行重新排列后，初始设定 32 位数据区块如下：

$$L_0 = t_1t_2\cdots t_{32} \quad R_0 = t_{33}t_{34}\cdots t_{64}$$

(2) 执行 16 轮迭代运算，第 i 轮迭代表达式如下($i=1,2,\cdots,16$)：

$$L_i \leftarrow R_{i-1}$$

$$R_i \leftarrow L_{i-1} \oplus f(R_{i-1}, K_i)$$

其中$\oplus$为异或运算，K_i(48 位)为密钥 K(56 位)所产生的子密钥。

(3) 执行 32 位数据区块的交换,亦即

$$L_{16} = c_1 c_2 \cdots c_{32} \leftarrow R_{16}$$

$$R_{16} = c_{33} c_{34} \cdots c_{64} \leftarrow L_{16}$$

(4) 令 $C = c_1 c_2 \cdots c_{64}$。将 C 进行逆初始置换,重新排列后,输出密文 C。

注:解密时密文 C 从 IP^{-1} 表进入,子密钥从 K_{16} 至 K_1 依序产生。

4.3.2 DES 详细的加密过程

(1) 初始置换表如表 4-2 所示,逆初始置换表如表 4-3 所示。

(2) f 函数 $f(R_{i-1}, K_i)$。

表 4-2 初始置换表 IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

表 4-3 逆初始置换表 IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

利用表 4-4 扩增排列表(简称表 E),将 R_{i-1}(32 位)扩充成 48 位,并将结果与 K_i 进行 XOR 运算形成 8 个区块,每一个区块为 6 位,亦即

$$B_1 = b_1 b_2 \cdots b_6 \quad B_2 = b_7 b_8 \cdots b_{12}$$

$$B_3 = b_{13} b_{14} \cdots b_{18} \quad B_4 = b_{19} b_{20} \cdots b_{24}$$

$$B_5 = b_{25} b_{26} \cdots b_{30} \quad B_6 = b_{31} b_{32} \cdots b_{36}$$

$$B_7 = b_{37} b_{38} \cdots b_{42} \quad B_8 = b_{43} b_{44} \cdots b_{48}$$

利用 S_i-box 将 B_i 置换成 4 位($i=1,2,\cdots,8$),将结果经过缩减排列表(简称表 P)排列,如表 4-5 所示,其结果为 $f(R_{i-1}, K_i)$。

表 4-4　扩增排列表 E

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

表 4-5　缩减排列表 P

16	17	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

(3) 利用 S_i-box 将 B_i 置换成 4 位($i=1,2,\cdots,8$)。

令 B_i 为 $x_1x_2\cdots x_6$。取 x_1x_6 之值为列坐标 r，取 $x_2x_3x_4x_5$ 之值为行坐标 c。

从表 4-6 替换盒(简称 S-box)的 S_i-box 中决定(r,c)之值，并以 4 位表示之。因为在 S_i-box 中的值都介于 0 至 15 之间，转成二进制时，仅需以 4 位表示之即可，故原来为 6 位之 B_i，经过 S_i-box 转换后变为 4 位。

例如，$(011001)_2$ 经由 S_3-box，由第 1 和第 6 位中得到 $(01)_2$，而第 2 至第 5 位得到 $(1100)_2$，即从第 1 列中的第 12 个位置得到数值 12，表示成二进制为 $(1100)_2$。

表 4-6　替换盒(S-box)

	行 列	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
S_1-box	0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S_2-box	0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3-box	0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4-box	0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

续表

	行 列	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S_5-box	0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S_6-box	0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7-box	0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8-box	0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

(4) 从密钥 K(56-bit)产生子密钥 K_i(48 位)。将每 7 位区块加上一个偶同位检查位，将 K 扩充成 64 位。利用表 4-7 的密钥排列-1 表将检查位除掉(因为在 PC-1 表中无 8,16,24,32,48,56,64)，则 K 又成为 56 位，表示为 PC-1(K)=CD，其中 C 与 D 各为 28 位。再经过表 4-9 左位移表(简称 LS)，将位做左旋的动作，表示为 $C_i=\mathrm{LS}_i(C_{i-1})$、$D_i=\mathrm{LS}_i(D_{i-1})$。利用表 4-8 密钥排列-2 表(简称 PC-2)将密钥转为 48 位，K_i=PC-2(C_iD_i)。

表 4-7　密钥排列 PC-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

表 4-8　密钥排列 PC-2

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	33	46	42	50	36	29	32

表 4-9　位移表 LS

迭代	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
位移数	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

4.4　分组密码的运行模式

和其他密码体制一样，DES 也存在安全性问题，即弱密钥与半弱密钥。所谓弱密钥是指在所有可能的密钥中，有某几个特别的密钥，会降低 DES 的安全性，所以使用者必须避免使用这几个弱密钥，换句话说，弱密钥是指密钥 K 全部为 1 或 0 或某一半全为 1 或全为 0，将会造成次密钥呈现方式 K_1 到 K_{16} 等同于 K_{16} 到 K_1，亦即密文再加密一次将会还原成明文，也就是不需要使用解密程序。所谓的半弱密钥是指一组密钥对 X 与 Y，使得以密钥 X 加密可以用另一密钥 Y 解密；反之亦然。

1999 年，美国 NIST 公布 FIPS PUB 46-3，该标准规范一个加强 DES 安全性的 3-DES 算法（或称为 Triple DES），如图 4-7 所示。

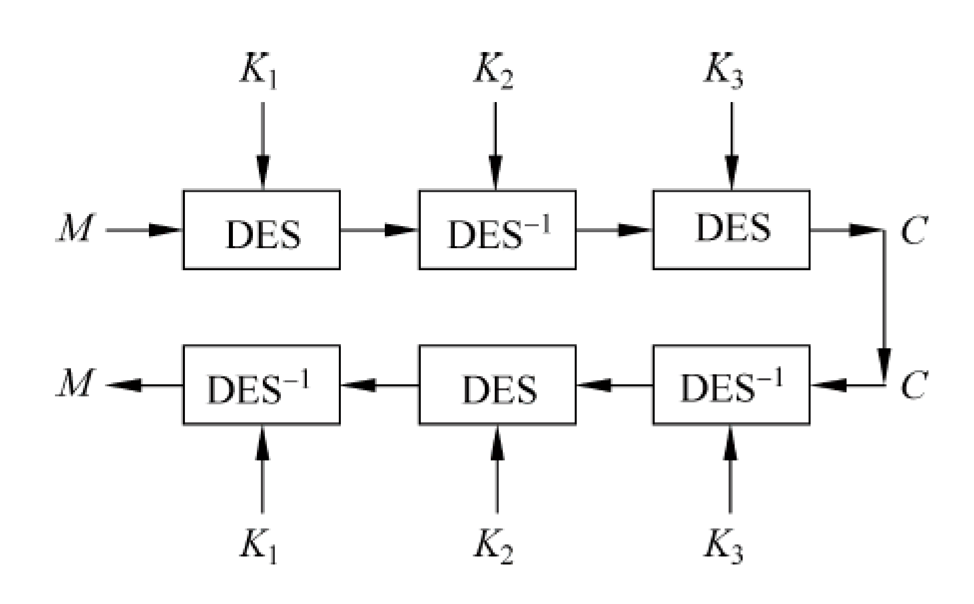


图 4-7　3-DES 加解密架构

3-DES 主要是由原本的 DES 加密与解密算法所组成的，其安全性更高且密钥长度更长。令 $E_K(M)=C$ 与 $D_K(C)=M$，其中 E 与 D 为 DES 加密与解密算法、K 为密钥、M 为明文以及 C 为密文。

3-DES 是三重数据加密算法（Triple Data Encryption Algorithm，TDEA）块密码的统称。它相当于对每个数据块应用三次 DES 加密算法。由于计算机运算能力的增强，原版 DES 密码的密钥长度变得容易被暴力破解；3-DES 是设计用来提供一种相对简单的方法，即通过增加 DES 的密钥长度来避免类似的攻击，而不是设计一种全新的块密码算法。

针对不同的应用，FIPS PUB 81 定义了区块加密法的 4 种操作模式：ECB 模式（Electronic Codebook Mode）、CBC 模式（Cipher Block Chaining Mode）、CFB 模式（Cipher Feedback Mode）以及 OFB 模式（Output Feedback Mode），此 4 种模式皆适用于 DES、3-DES 及 AES。此 4 个模式的运作情形分述如下。

1. ECB 模式

每一个明文区块皆使用同一把密钥来加密或解密，且加密或解密每一个区块的动作是

彼此独立的，亦即相同的明文区块经过 DES 加密后所产生的密文区块亦会相同，如图 4-8 所示。

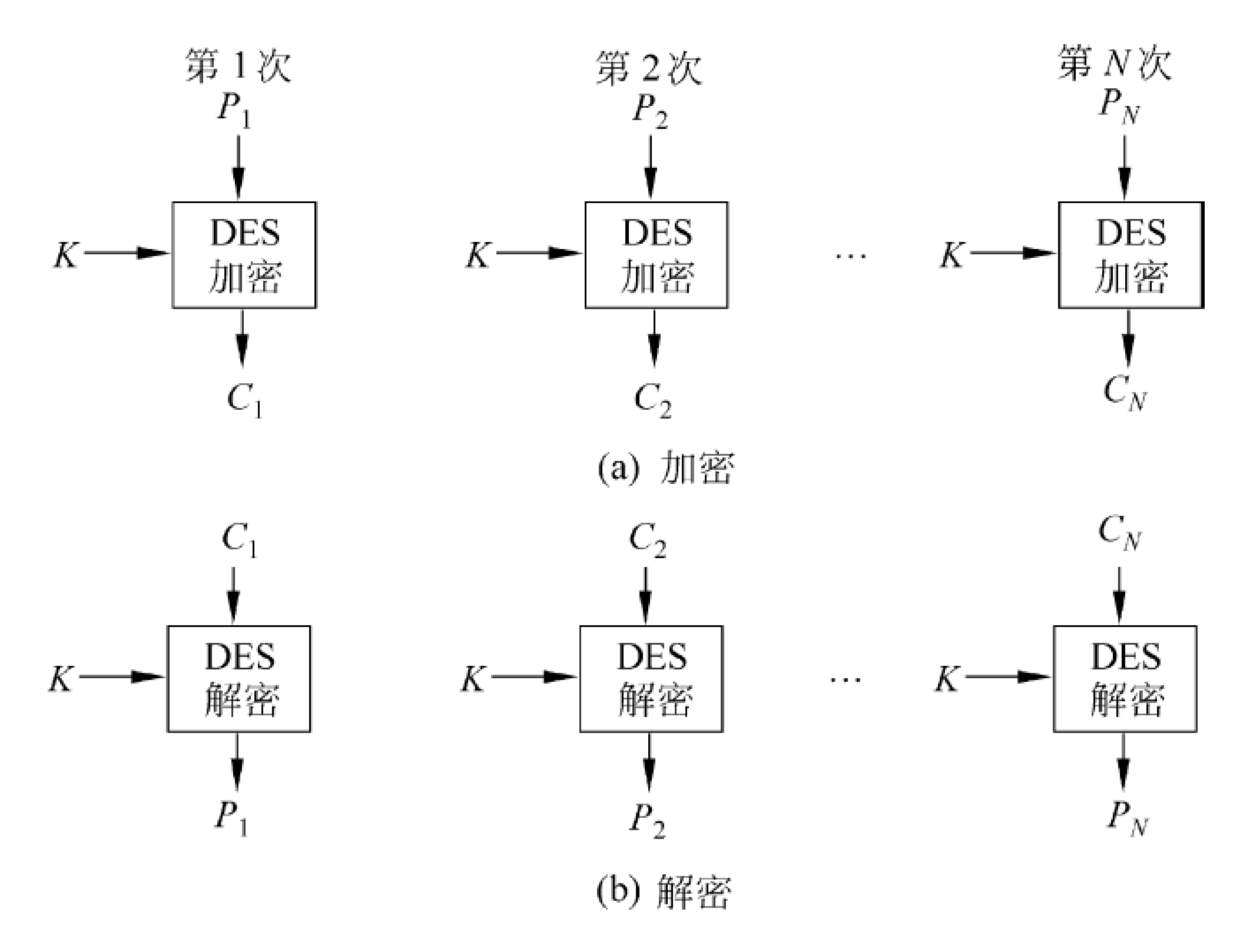


图 4-8 ECB 模式

ECB 在用于短数据(如加密密钥)时非常理想，因此如果需要安全地传递 DES 密钥，ECB 是最合适的模式。ECB 的最大特性是同一明文分组在消息中重复出现的话，产生的密文分组也相同。

ECB 用于长消息时可能不够安全，如果消息有固定结构，密码分析者有可能找出这种关系。例如，已知消息总是以某个预定义字段开始的，那么分析者就可能得到很多明文密文对。如果消息有重复的元素而重复的周期是 64 的倍数，那么密码分析者就能够识别这些元素。以上这些特性都有助于密码分析者，有可能为其提供对分组的代换或重排的机会。

2. CBC 模式

利用链接特性加强区块彼此的关系是相依的而非独立的。加密每一数据区块前，该数据区块必须先与前一次所产生的密文区块进行异或运算，之后再利用 DES 加密该异或结果。解密时，先针对每一区块密文做 DES 解密动作，之后再与前一个密文区块进行异或运算，此模式如图 4-9 所示。由于 CBC 模式的链接机制，CBC 模式对加密长于 64 比特的消息非常合适。

3. CFB 模式

加密每一数据区块时，必须先针对前一个所产生的密文区块(当加密第一个资料区块时，则为初始值 IV)进行 DES 加密，之后再与数据区块进行异或运算便可获得此区块的密文。反之，解密时亦必须先针对前一个密文区块(当加密第一个资料区块时，则为初始值 IV)进行 DES 解密，之后再与欲解密的密文区块进行异或运算，便可恢复此区块的明文，此模式如图 4-10 所示。

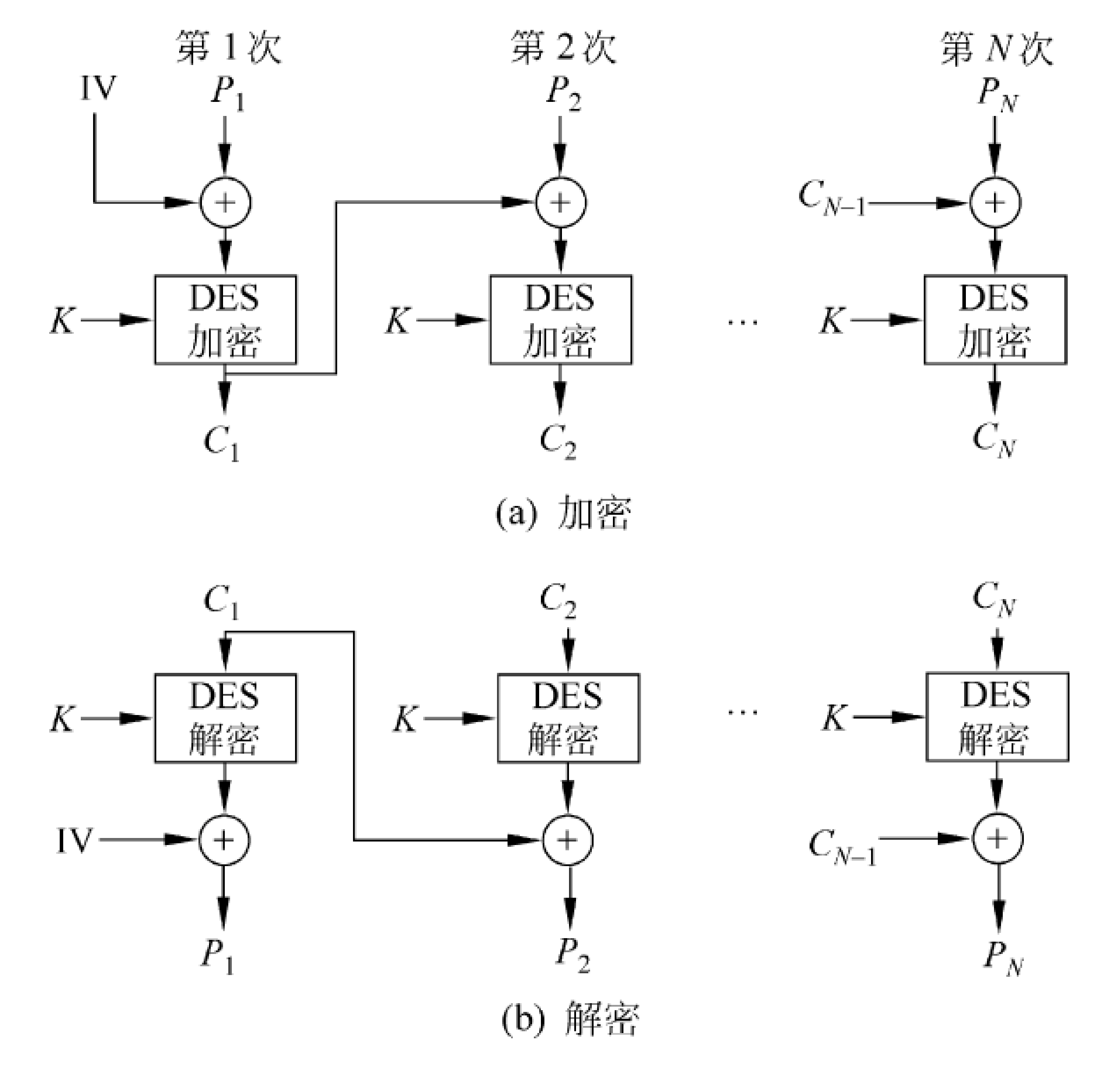


图 4-9　CBC 模式

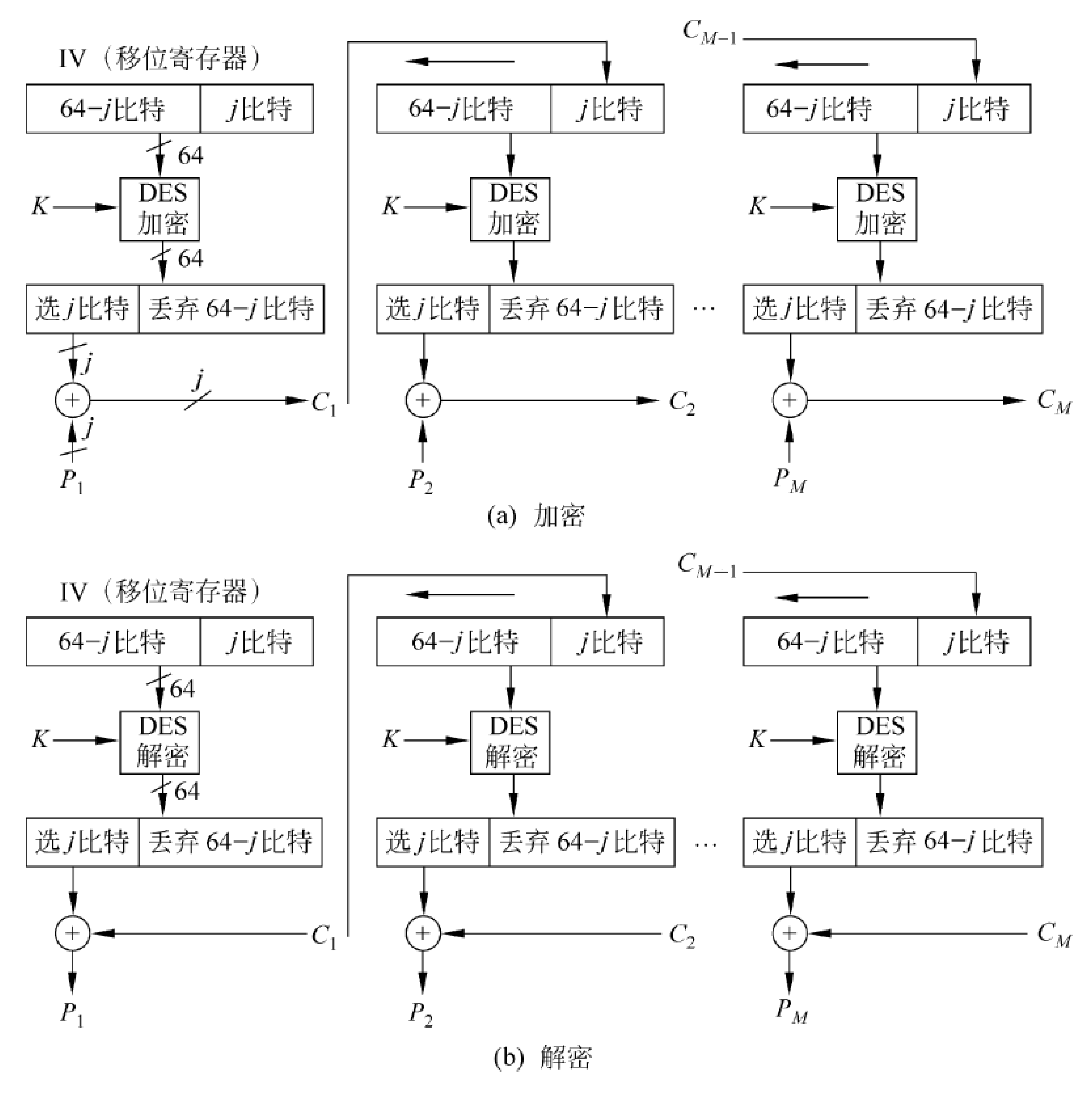


图 4-10　CFB 模式

4. OFB 模式

欲加密每一数据区块时，必须先针对前一个加密动作所使用的 DES 输出值（当加密第一个数据区块时，则为初始值 IV）进行 DES 运算以产生一个输出值，该值再与此欲加密的数据区块进行异或运算，便可获得此区块的密文。反之，解密时亦必须先针对前一个解密动作所使用的 DES 输出值（当解密第一个数据区块时，则为初始值 IV）进行 DES 运算以得到一个输出值，该值再与欲解密的密文区块进行异或运算以恢复此区块的明文，如图 4-11 所示。OFB 与 CFB 运作模式相当类似，差别在于异或运算的数据有所不同。就信息网络观点而言，OFB 模式优于 CFB 模式在于错误的位不会影响其他加密或解密的结果。就信息安全观点而言，OFB 模式比 CFB 模式易受修改攻击。

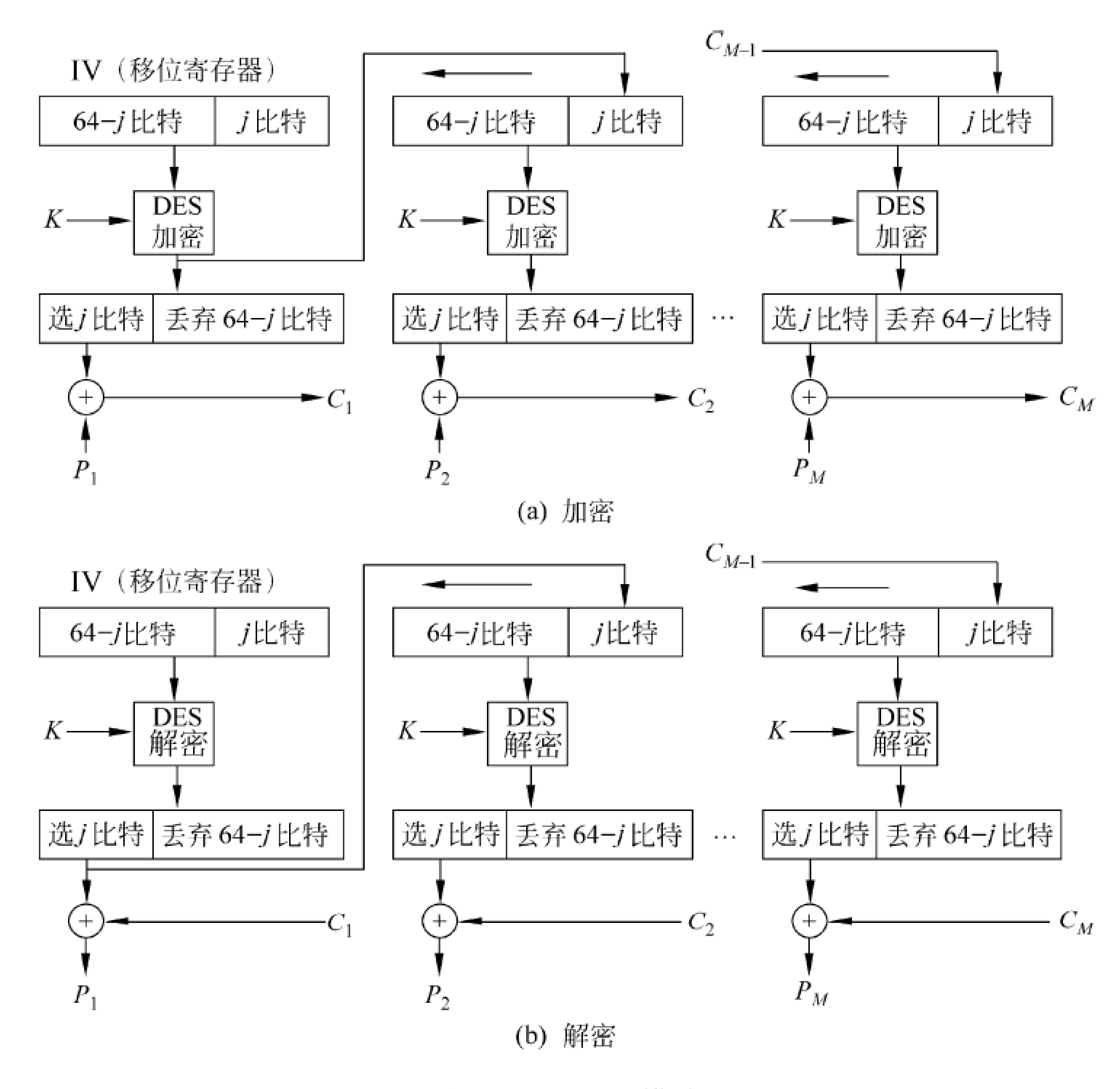


图 4-11 OFB 模式

4.5 DES 密码分析

DES 的加密单位仅有 64 位二进制，而且其中某些位还要用于奇偶校验或其他通信开销，有效密钥只有 56 位，这种特性必然降低了密码体制的安全性。因此，人们会对 56 位密钥的安全性产生质疑，那么 56 位密钥是否足够，已成为人们争论的焦点之一。

在 DES 算法中存在 12 个半弱密钥和 4 个弱密钥。由于在子密钥的产生过程中，密钥被分成了两个部分，如果这两个部分分成了全 0 或全 1，那么每轮产生的子密钥都是相同的，当密钥是全 0 或全 1，或者一半是 1 或 0 时，就会产生弱密钥或半弱密钥，DES 算法的安

全性就会变弱。在设定密钥时应避免弱密钥或半弱密钥的出现。

4.5.1　密码分析方法

虽然已发表的针对DES的密码分析的研究文章多于所有其他的分组密码，到目前为止，最实用的攻击方法仍然是暴力攻击。已知DES有一些次要的可能导致加密强度降低的密码学特性，同时有三种理论攻击的理论复杂性小于暴力破解，但需要不现实的已知明文或选择明文数量，并无实用价值。

对于一切密码而言，最基本的攻击方法是暴力破解法，即依次尝试所有可能的密钥。密钥长度决定了可能的密钥数量，因此也决定了这种方法的可行性。对于DES，即使在它成为标准之前就有一些关于其密钥长度的适当性问题，而且也正是它的密钥长度，而不是理论密码分析迫使它被后续算法所替代。在设计时，在与包括NSA在内的外部顾问讨论后，密钥长度被从128位减少到了56位以适应在单芯片上实现算法。

在学术上，曾有数个DES破解器被提出。1977年，迪菲和海尔曼提出了一部造价约2千万美元的破解器，可以在一天内找到一个DES密钥。1993年，迈克尔·维纳设计了一部造价约1百万美元的破解器，大约可以在7小时内找到一个密钥。然而，这些早期的设计并没有被实现，至少没有公开地实现。在20世纪90年代晚期，DES开始受到实用性的攻击。1997年，RSA安全赞助了一系列竞赛，奖励第一个成功破解以DES加密的信息的队伍1万美元，洛克·韦尔谢什(Rocke Verser)、马特·柯廷(Matt Curtin)和贾斯廷·多尔斯基(Justin Dolske)领导的DESCHALL计划获胜，该计划使用了数千台连接到互联网的计算机的闲置计算能力。1998年，电子前哨基金会(EFF，一个信息人权组织)制造了一台DES破解器，造价约25万美元，如图4-12所示。该破解器包括1856个自定义芯片，可以用稍多于2天的时间暴力破解一个密钥，它显示了迅速破解DES的可能性。EFF的动力来自于向大众显示DES不仅在理论上，在使用上也是可破解的。

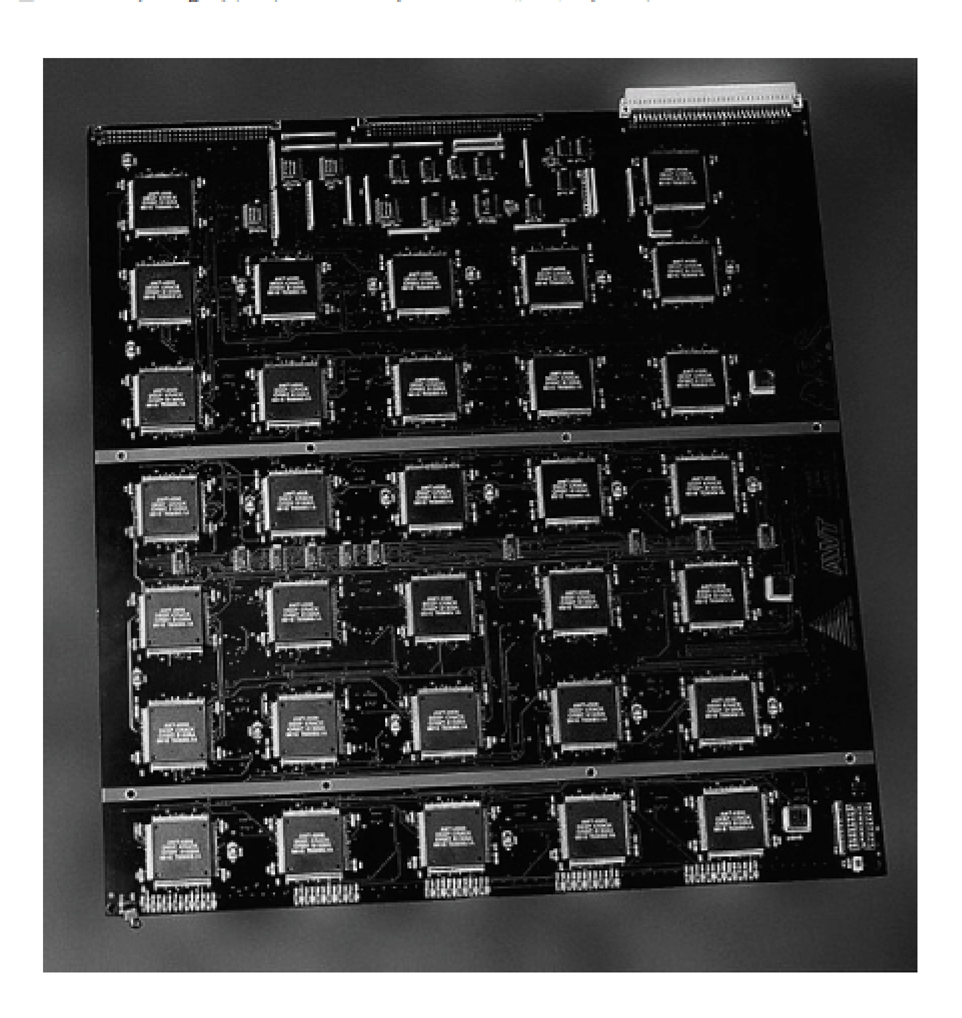


图4-12　价值250 000美元的DES破解器

下一个确认的DES破解器是2006年由德国的鲁尔大学与基尔大学的工作组建造的COPACOBANA。与EFF的不同，COPACOBANA由商业上可获得的、可重配置的FPGA

组成。120 片并行的 XILINX Spartan3-1000 型 FPGA 分为 20 个 DIMM 模块,每个模块包括 6 个 FPGA。使用可重配置的 FPGA 使得这种设备也可以用于其他密码的破解。另外一个关于 COPACOBANA 的有趣事实是它的成本。一台 COPACOBANA 的造价大约是 1 万美元,是 EFF 设备的 25 分之一,这充分说明了数字电路的持续进步。考虑到通货膨胀因素,同样价格的设备的性能在 8 年间大约提到了原来的 30 倍。2007 年,COPACOBANA 的两个项目参与者组建的 SciEngines 公司改进了 COPACOBANA,并发展了它的下一代。2008 年,他们的 COPACOBANA RIVYERA 将破解 DES 的时间减少到了 1 天以内,使用 128 片 Spartan-3 5000 型 FPGA。目前 SciEngines 的 RIVYEAR 保持着使用暴力破解法破解 DES 的纪录。

有三种已知方法可以用小于暴力破解的复杂性,去破解 DES 的全部 16 回次:微分密码分析(DC)、线性密码分析(LC)以及戴维斯攻击。然而,这些攻击都是理论性的,难以用于实践;它们有时被归结于认证的弱点。

1. 微分密码分析

在 20 世纪 80 年代晚期由艾力·毕汉姆和阿迪·萨莫尔重新发现;20 世纪 70 年代 IBM 和 NSA 便发现了这种方法,但没有公开。为了破解全部 16 回次,微分密码分析需要 2^{47} 组选择明文。DES 被设计为对 DC 具有抵抗性。

2. 线性密码分析

由松井充(Mitsuru Matsui)发现,需要 2^{43} 组已知明文;该方法已被实现,是第一种公开的实验性的针对 DES 的密码分析。没有证据显示 DES 的设计可以抵抗这种攻击方法。一般概念上的 LC——“多线性密码分析”——在 1994 年由 Kaliski 和 Robshaw 所建议,并由比留科夫等人于 2004 年所改进。线性密码分析的选择明文变种是一种类似的减少数据复杂性的方法。帕斯卡尔·朱诺德(Pascal Junod)在 2001 年进行了一些确定线性密码分析的实际时间复杂性的实验,结果显示它比预期的要快,需要约 $2^{39}-2^{41}$ 次操作。

3. 改进的戴维斯攻击

线性和微分密码分析是针对很多算法的通用技术,而戴维斯攻击是一种针对 DES 的特别技术,在 20 世纪 80 年代由唐纳德·戴维斯(Donald Davies)首先提出,并于 1997 年为毕汉姆和亚历克斯·比留科夫(Alex Biryukov)所改进。其最有效的攻击形式需要 2^{50} 组已知明文,计算复杂性也为 2^{50},成功率为 51%。

也有一些其他的针对削减了回次的密码版本,即少于 16 回次的 DES 版本。这些攻击显示了多少回次是安全所需的,以及完整版本拥有多少“安全余量”。微分线性密码分析于 1994 年为兰福德(Langford)和海尔曼所提出,是一种组合了微分和线性密码分析的方法。一种增强的微分线性密码分析版本可以利用 $2^{15.8}$ 组已知明文以 $2^{29.2}$ 的时间复杂性破解 9 回次的 DES。

接下来以线性密码分析为例进行说明。

4.5.2 线性密码分析

线性分析的分析者利用了包含明文、密文和子密钥的线性表达式进行分析。

线性分析前提：假设攻击者已经知道了大量的明文和相对应的密文。

线性分析最基本的思想就是用一个线性表达式来近似表示加密算法的一部分，该线性表达式是关于模 2 的操作(用⊕表示 XOR 操作)。表达式具有以下的形式：

$$X_{i1} \oplus X_{i2} \oplus \cdots \oplus X_{iu} \oplus X_{j1} \oplus X_{j2} \oplus \cdots \oplus X_{jv} = 0$$

该表达式是 u 比特的输入与 v 比特的输出进行 XOR 操作的结果。

线性密码分析的方法就是测定上述形式的线性表达式发生的可能性的大小。

如果一个密码算法使得等式成立的可能性非常大或者非常小，这说明该密码算法的随机性比较差。一般情况下，假如随机选择 $u+v$ 个比特值，并且将它们代入等式，等式成立的可能性应该为 1/2。在线性密码分析中，一个线性表达式成立的可能性与 1/2 之间的差值定义为偏移量(或偏差)。一个线性表达式成立的可能性与 1/2 的距离越大，密码分析者利用线性密码分析就越有效果。如果对于随机选择的明文找到相应的密文，使得上述表达式成立的可能性为 PL，则线性可能性偏移量是 PL－1/2。线性可能性偏移量的绝对值 |PL－1/2| 越大，线性密码分析对于知道较少明文情况下的攻击越有效。

考虑两个二进制变量 X_1, X_2。通过计算简单的关系开始：

$X_1 \oplus X_2=0$ 是一个线性表达式，等价于 $X_1=X_2$；

$X_1 \oplus X_2=1$ 是一个仿射表达式，等价于 $X_1 \neq X_2$。

给出以下的概率分布：

$$\Pr(X_1 = i) = \begin{cases} p_1 & i = 0 \\ 1 - p_1 & i = 1 \end{cases}$$

$$\Pr(X_2 = i) = \begin{cases} p_2 & i = 0 \\ 1 - p_2 & i = 1 \end{cases}$$

如果两个随机变量相互独立，则

$$\Pr(X_1 = i, X_2 = j) = \begin{cases} p_1 p_2 & i = 0, j = 0 \\ p_1(1 - p_2) & i = 0, j = 1 \\ (1 - p_1)p_2 & i = 1, j = 0 \\ (1 - p_1)(1 - p_2) & i = 1, j = 1 \end{cases}$$

可以等价表示为

$$\begin{aligned}\Pr(X_1 \oplus X_2 = 0) &= \Pr(X_1 = X_2) \\ &= \Pr(X_1 = 0, X_2 = 0) + \Pr(X_1 = 1, X_2 = 1) \\ &= p_1 p_2 + (1 - p_1)(1 - p_2)\end{aligned}$$

另一种表示方法是：

令 $p_1=1/2+\varepsilon_1, p_2=1/2+\varepsilon_2$，$\varepsilon_1$、$\varepsilon_2$ 是线性可能性偏移量，而且 $-1/2\leqslant\varepsilon_1, \varepsilon_2\leqslant 1/2$。

因此，$\Pr(X_1 \oplus X_2=0)=1/2+2\varepsilon_1\varepsilon_2$，并且 $X_1 \oplus X_2=0$ 的线性偏移量 $\varepsilon_{1,2}$ 为 $2\varepsilon_1\varepsilon_2$。

扩展到多个随机二进制变量的情况：

若有随机变量 $X_1 \cdots X_n$ 且概率为 $p_1=1/2+\varepsilon_1, p_2=1/2+\varepsilon_2, \cdots, p_n=1/2+\varepsilon_n$。则

$\Pr(X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0) = 1/2 + 2\varepsilon_1\varepsilon_2\varepsilon_n$。

Piling-Up 引理 对于 n 个相互独立的二进制随机变量，$X_1, X_2, \cdots, X_n$，可得

$$\Pr(X_1 \oplus \cdots \oplus X_n = 0) = 1/2 + 2^{n-1}\prod_{i=1}^{n}\varepsilon_i$$

或者等价表示为

$$\Pr(X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0) = 1/2 + 2^{n-1}\varepsilon_1\varepsilon_2\cdots\varepsilon_n$$

其中，ε_1、ε_2、ε_n表示 $X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0$ 的线性偏移量。

举例说明。

有 4 个相互独立的随机变量 X_1, X_2, X_3 和 X_4。

$\Pr(X_1 \oplus X_2 = 0) = 1/2 + \varepsilon_{1,2}$

$\Pr(X_2 \oplus X_3 = 0) = 1/2 + \varepsilon_{2,3}$

$\Pr(X_1 \oplus X_3 = 0) = \Pr([X_1 \oplus X_2] \oplus [X_2 \oplus X_3] = 0)$

应用 Piling-Up 引理可得

$\Pr(X_1 \oplus X_3 = 0) = 1/2 + 2\varepsilon_{1,2}\varepsilon_{2,3}$

相应地，$\varepsilon_{1,3} = 2\varepsilon_{1,2}\varepsilon_{2,3}$。

下面针对 DES 算法进行攻击性分析。

在更详细地讨论攻击细节之前，首先需要知道 S-box 盒的线性脆弱性。S-box 的输入为 $\boldsymbol{X} = [X_1\ X_2\ X_3\ X_4]$，相应的输出为 $\boldsymbol{Y} = [Y_1\ Y_2\ Y_3\ Y_4]$，如图 4-13 所示。

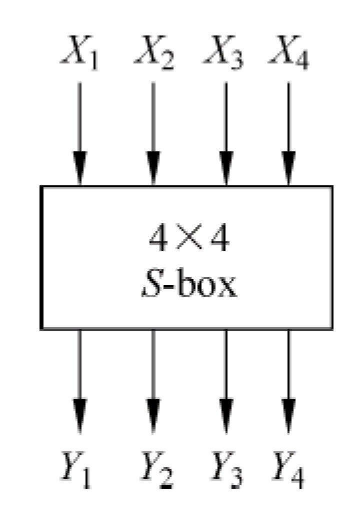


图 4-13 S-box 的输入和输出

根据对 S-box 的线性近似采样，对于 16 种可能的输入 $\boldsymbol{X}$ 和其相应的输出 $\boldsymbol{Y}$，有 12 种情况可以使得 $X_2 \oplus X_3 \oplus Y_1 \oplus Y_3 \oplus Y_4 = 0$ 或等价形式 $X_2 \oplus X_3 = Y_1 \oplus Y_3 \oplus Y_4$ 成立，因此线性可能性偏移量是 $12/16 - 1/2 = 1/4$。

相似地，对于等式 $X_1 \oplus X_4 = Y_2$，其线性可能性偏移量接近于 0，而等式 $X_3 \oplus X_4 = Y_1 \oplus Y_4$ 的线性可能性偏移量是 $2/16 - 1/2 = -3/8$。

S-box 线性近似采样如表 4-10 所示。

表 4-10 S-box 线性近似采样

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4	$X_2 \oplus X_3$	$Y_1 \oplus Y_3 \oplus Y_4$	$X_1 \oplus X_4$	Y_2	$X_3 \oplus X_4$	$Y_1 \oplus Y_4$
0	0	0	0	1	1	1	0	0	0	0	1	0	1
0	0	0	1	0	1	0	0	0	0	1	1	1	0
0	0	1	0	1	1	0	1	1	0	0	1	1	0
0	0	1	1	0	0	0	1	1	1	1	0	0	1
0	1	0	0	0	0	1	0	1	1	0	0	0	0
0	1	0	1	1	1	1	1	1	1	1	1	1	0
0	1	1	0	1	0	1	1	0	1	0	0	1	0
0	1	1	1	1	0	0	0	0	1	1	0	0	1
1	0	0	0	0	0	1	1	0	0	1	0	0	1
1	0	0	1	1	0	1	0	0	0	0	0	1	1
1	0	1	0	0	1	1	0	1	1	1	1	1	0
1	0	1	1	1	1	0	0	1	1	0	1	0	1

续表

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4	$X_2 \oplus X_3$	$Y_1 \oplus Y_3 \oplus Y_4$	$X_1 \oplus X_4$	Y_2	$X_3 \oplus X_4$	$Y_1 \oplus Y_4$
1	1	0	0	0	1	0	1	1	1	1	1	0	1
1	1	0	1	1	0	0	1	1	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1	0	1	0
1	1	1	1	0	1	1	1	0	0	0	1	0	1

例如，一个输入变量的线性近似表达式 $a_1 \cdot X_1 \oplus a_2 \cdot X_2 \oplus a_3 \cdot X_3 \oplus a_4 \cdot X_4$，其中，$a_i \in \{0,1\}$。"·"为二进制的"与"运算，输入行的十六进制的值是 $a_1\ a_2\ a_3\ a_4$ 的组合。

相似地，对于一个输出变量的线性近似表达式 $b_1 \cdot Y_1 \oplus b_2 \cdot Y_2 \oplus b_3 \cdot Y_3 \oplus b_4 \cdot Y_4$，其中，$b_i \in \{0,1\}$，输出行的十六进制的值是 $b_1 b_2 b_3 b_4$ 的组合。

其中，Input 表示表达式的输入系数，而 output 表示表达式的输出系数，行和列交集处的值表示以此行列值代表线性表达式成立的数量减去 8，线性近似偏移量如表 4-11 所示。

表 4-11　线性近似偏移量

		Output															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	+8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	−2	−2	0	0	−2	+6	+2	+2	0	0	+2	+2	0	0
	2	0	0	−2	−2	0	0	−2	−2	0	0	+2	+2	0	0	−6	+2
	3	0	0	0	0	0	0	0	0	+2	−6	−2	−2	+2	+2	−2	−2
	4	0	+2	0	−2	−2	−4	−2	0	0	−2	0	+2	+2	−4	+2	0
	5	0	−2	−2	0	−2	0	+4	+2	−2	0	−4	+2	0	−2	−2	0
	6	0	+2	−2	+4	+2	0	0	+2	0	−2	+2	+4	−2	0	0	−2
Input	7	0	−2	0	+2	+2	−4	+2	0	−2	0	+2	0	+4	+2	0	+2
	8	0	0	0	0	0	0	0	0	−2	+2	+2	−2	+2	−2	−2	−6
	9	0	0	−2	−2	0	0	−2	−2	−4	0	−2	+2	0	+4	+2	−2
	A	0	+4	−2	+2	−4	0	+2	−2	+2	+2	0	0	+2	+2	0	0
	B	0	+4	0	−4	+4	0	+4	0	0	0	0	0	0	0	0	0
	C	0	−2	+4	−2	−2	0	+2	0	+2	0	+2	+4	0	+2	0	−2
	D	0	+2	+2	0	−2	+4	0	+2	−4	−2	+2	0	+2	0	0	+2
	E	0	+2	+2	0	−2	−4	0	+2	−2	0	0	−2	−4	+2	−2	0
	F	0	−2	−4	−2	−2	0	+2	0	0	−2	+4	−2	−2	0	+2	0

如线性表达式 $X_3 \oplus X_4 = Y_1 \oplus Y_4$（输入行标是 3，输出列标是 9），表中值−6，代表线性表达式成立的数量为 2 次。

4.6　高级加密标准

4.6.1　AES 概述

DES 作为密码学上的一个里程碑，尽管在安全上是脆弱的，但由于快速 DES 芯片的大

量生产，使得 DES 仍能暂时继续使用，为提高安全强度，通常使用独立密钥的三级 DES，同时急需一种新的密码体制来代替 DES。

NIST 于 1997 年 4 月正式公告下一代加密标准(Advanced Encryption Standard，AES)的征选。制定 AES 的主要目标是确保数据可达到 100 年的安全性，即加密后的密文在 100 年内不会被破解，因此安全性与运算效率皆须在 Triple-DES 之上，NIST 亦规划在未来的 30 年内将逐渐推广政府及民间企业间通用的资料加密标准由 DES 更换为 AES。在经过三个回合的技术分析会议后(表 4-12 为 AES 在发展期间的重大纪事)，从 15 个候选算法中(表 4-13 与表 4-14 分别为 AES 在第一回合 15 个候选算法的基本数据与比较)，于 2000 年 10 月 2 日公开选定由比利时 J. Daemen 与 V. Rijmen 两位学者所设计的 Rijndael 算法为 AES 所用。

表 4-12　AES 的历史

时　间	事　件
1997 年 7 月	公告 AES 之征选
1998 年 4 月 15 日	开始接受 AES 候选算法
1998 年 7 月 15 日	结束接受 AES 候选算法
1998 年 8 月 20～22 日	开始 15 个 AES 候选算法第一回合的技术分析
1999 年 3 月 22～23 日	举行 AES 第二次会议
1999 年 4 月 15 日	结束 AES 第一回合的技术分析，并选出 5 个第二回合的技术分析候选算法：MARS、RC6、Rijndael、Serpent 及 Twofish
2000 年 8 月 13～14 日	结束第二回合的技术分析
2000 年 5 月 15 日	发表 5 个候选算法的公开评论与分析
2000 年 10 月 2 日	公开选定 Rijndael 为 AES 所采用
2000 年 11 月	公布 Rijndael 为 AES FIPS 草案，并接受 90 天的公开评论
2001 年 2 月 28 日	公布对 Rijndael 的公开评论并开始修正草案
2001 年 11 月 26 日	公告 FIPS PUB 197，正式成为官方新一代通行的加密标准，并完成相关测试

表 4-13　15 个 AES 候选算法

算　法	提案人或公司	国　家
CAST-256	Entrust Inc.	Canada
Crypton	Future System Inc.	Korea
Deal	Richard Outerbridge，Lars Knudsem	Canada
DFC	CNRS-Centre National pour la Recherche Scientifique	France
E2	NTT	Japan
Frog	TecApro International S. A.	Costa Rica
HPC	Rich Schroeppel	USA
LOKI97	Lawrie Brown，Josef Pieprzyk，Jennifer Seberry	Australia
Megenta	Deutsche Telekom	Germany
Mars	IBM	USA
RC6	RSA	USA
Rijndael	Joan Deamen，Vincent Rijmen	Belgium
Safer＋	Cylink Corp.	USA
Serpent	Ross Anderson，Eli Biham，Lars Knudsen	UK，Israel，Norway
Twofish	Counterpane	USA

表 4-14　NIST 对 AES 之 15 个候选算法比较

算　法	区块长度/位	密钥长度/位	结　构	回合数	最小回合数
CAST-256	128	128～256	Ext. Feistel Network	48	40
Crypton	128	256	Square	12	11
Deal	128	128,192,256	Feistel Network	6,8,8	10
DFC	128	256	Feistel Network	8	8
E2	128	128,192,256	Feistel Network	12	10
Frog	64～1024	40～1000	Special	8	8
HPC	Any	Any	Omni	8	8
LOKI97	128	128,192,256	Feistel Network	16	38
Megenta	128	128	Feistel Network	6,8,8	11
Mars	128	128～1248	Ext. Feistel Network	32	20
RC6	128	256	Feistel Network	20	21
Rijndael	128,192,256	128,192,256	Square	10,12,16	8
Safer＋	128	128,192,256	SP Network	8,12,16	7
Serpent	128	256	SP Network	32	17
Twofish	128	128,192,256	Feistel Network	16	14

Rijndael 算法除具备低成本、高安全性特性外，最大的优点在于即使在受限的工作环境下，如较小的内存空间中，仍有很好的加解密运算效率；而运算子的设计，亦容易抵抗完全搜寻攻击，如此便能保证 AES 可有较长的安全周期。在经过一连串的公开评论与测试后，NIST 于 2001 年 11 年 26 日发布 FIPS PUB 197，正式将 AES 定为美国新一代的资料加密标准。

AES 为一个区块加密法，在原先 Rijndael 的设计中，允许可变动的数据区块及密钥的长度，且数据区块与密钥长度的变动是各自独立的，密钥长度与数据长度有 128、192 及 256 位的选择，而运算回合数由密钥及明文(数据区块)长度而定，因此对于任何系统都可以弹性地运用。但在 AES 标准中，将数据(明文/密文)长度固定为 128 位，而依密钥长度不同 AES 可分为 AES-128、AES-192 以及 AES-256，其密钥长度、数据长度和运算回合数的关系如表 4-15 所示。

表 4-15　AES 密钥长度、数据长度与运算回合数对照表

	密钥长度(N_k words)	数据长度(N_b words)	运算回合数(N_r)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

4.6.2　AES 中的数学基础

定义 4-1　设 G 为非空集合，在 G 内定义了一种代数运算，若满足下述公理，则 G 构成一个群。

(1) 闭合：对于所有 G 中的 a、b，运算 $a \cdot b$ 的结果也在 G 中。

(2) 结合律：对于所有 G 中的 a、b 和 c，等式 $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ 成立。

(3) 单位元：存在 G 中的一个元素 e，使得对于所有 G 中的元素 a，等式 $e \cdot a = a \cdot e = a$ 成立。

(4) 逆元：对于每个 G 中的 a，存在 G 中的一个元素 b 使得 $a \cdot b = b \cdot a = e$，这里的 e 是单位元。

若群 G 满足交换律，则称群 G 为交换群或阿贝尔群。

定义 4-2 非空集合元素 F，在 F 中定义了加和乘两种运算，若满足下述公理，则称 F 为一个域。

(1) F 关于加法构成阿贝尔群。

(2) F 中非零元素全体对乘法构成阿贝尔群，其乘法恒等元(单位元)记为 1。

(3) 加法和乘法间的分配律：$a(b+c)=ab+ac$ 和 $(b+c)a=ba+ca$。

若 F 中的元素为有限个，称 F 为有限域(Finite Field)。

定义 4-3 对于多项式 $f(x)$，设它的次数为 n，表示为 $\deg(f)=n$。对于多项式 $f(x)$、$g(x)$、$h(x)$，设 $\deg(g)=n$，如果 $g(x)=q(x)f(x)+h(x)$，其中 $\deg(h)<n$，则可定义 $g(x)\equiv h(x) \bmod f(x)$，即多项式同余。

定义 4-4 有限域是指仅含有限多个元素的域。

有限域中的元素可以用多种不同的方式表示。对于任意素数的方幂，都有唯一的一个有限域，因此 GF(2^8)的所有表示是同构的，但不同的表示方法会影响到 GF(2^8)上运算的复杂度。

将 $b_7b_6b_5b_4b_3b_2b_1b_0$ 构成的字节 b 看成系数在$\{0,1\}$中的多项式。

$$b_7x^7+b_6x^6+b_5x^5+b_4x^4+b_3x^3+b_2x^2+b_1x+b_0$$

例如：十六进制数 57 对应的二进制为 01010111，看成一个字节，对应的多项式为

$$x^6+x^4+x^2+x+1$$

在多项式表示中，GF(2^8)上两个元素的和仍然是一个次数不超过 7 的多项式，其系数等于两个元素对应系数的模 2 加(比特异或)。

例如：57+83=D4，用多项式表示为

$$(x^6+x^4+x^2+x+1)+(x^7+x+1)=x^7+x^6+x^4+x^2(\bmod m(x))$$

用二进制表示为

$$01010111+10000011=11010100$$

由于每个元素的加法逆元等于自己，所以减法和加法相同。

要计算 GF(2^8)上的乘法，必须先确定一个 GF(2)上的 8 次不可约多项式；GF(2^8)上两个元素的乘积就是这两个多项式的模乘(以这个 8 次不可约多项式为模)。在 Rijndael 密码中，这个 8 次不可约多项式确定为

$$m(x)=x^8+x^4+x^3+x+1$$

它的十六进制表示为 11B。

例如，57 · 83=C1 可表示为以下的多项式乘法：

$$(x^6+x^4+x^2+x+1)\cdot(x^7+x+1)=x^7+x^6+1(\bmod m(x))$$

乘法运算虽然不是标准的按字节的运算，但也是比较简单的计算部件。

AES 主要是使用 GF(2^8)作为字节阶层运算的基础架构及采用 4 字节的字组运算。

4.6.3　AES 算法

AES 的原理框图如图 4-14 所示。算法主要由 4 个不同的计算部件组成，分别是字节代换(ByteSub)、行移位(ShiftRow)、列混合(MixColumn)、密钥加(AddRoundKey)。

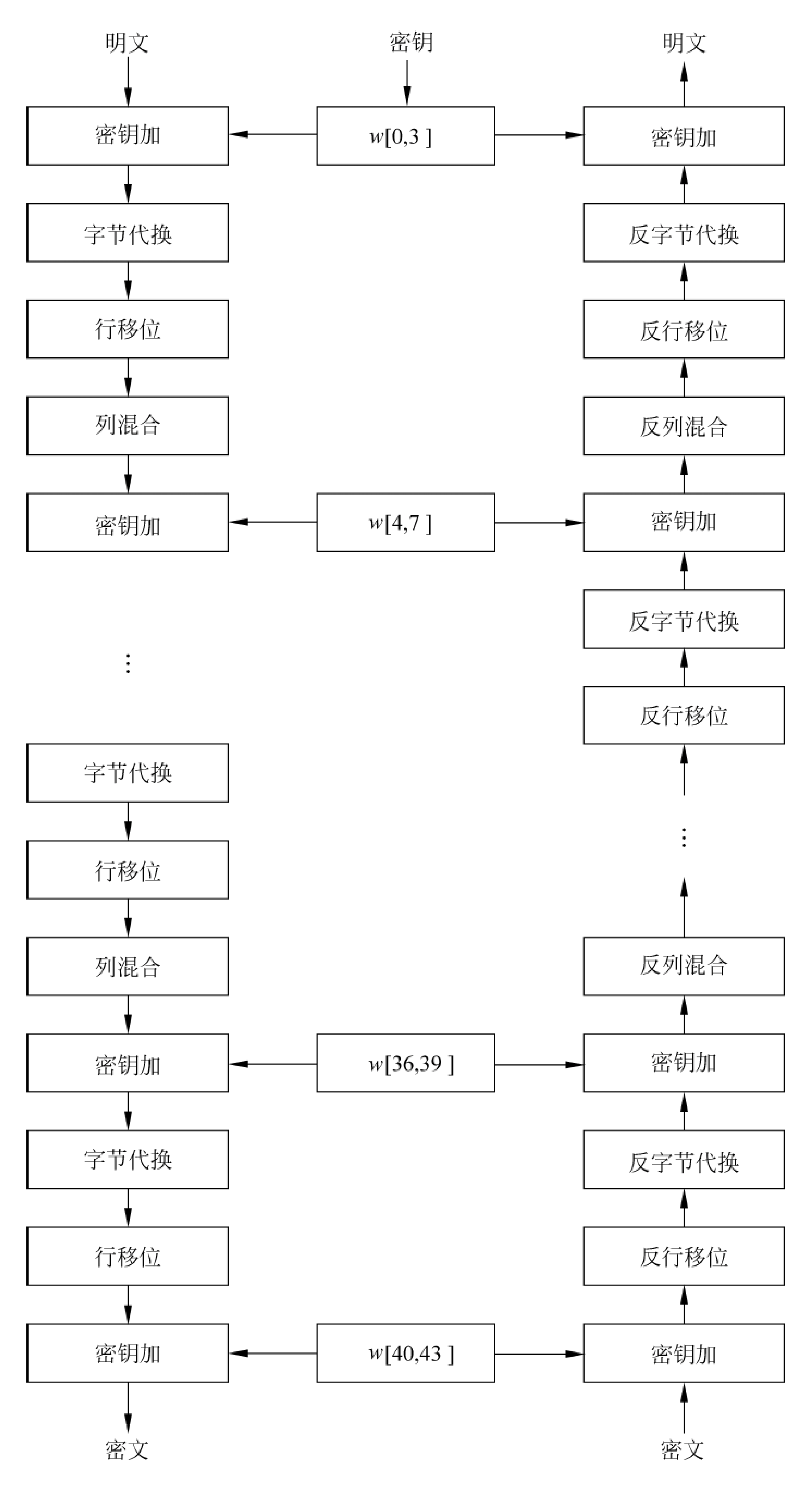


图 4-14　AES 的原理图

加密流程如图 4-15 所示。

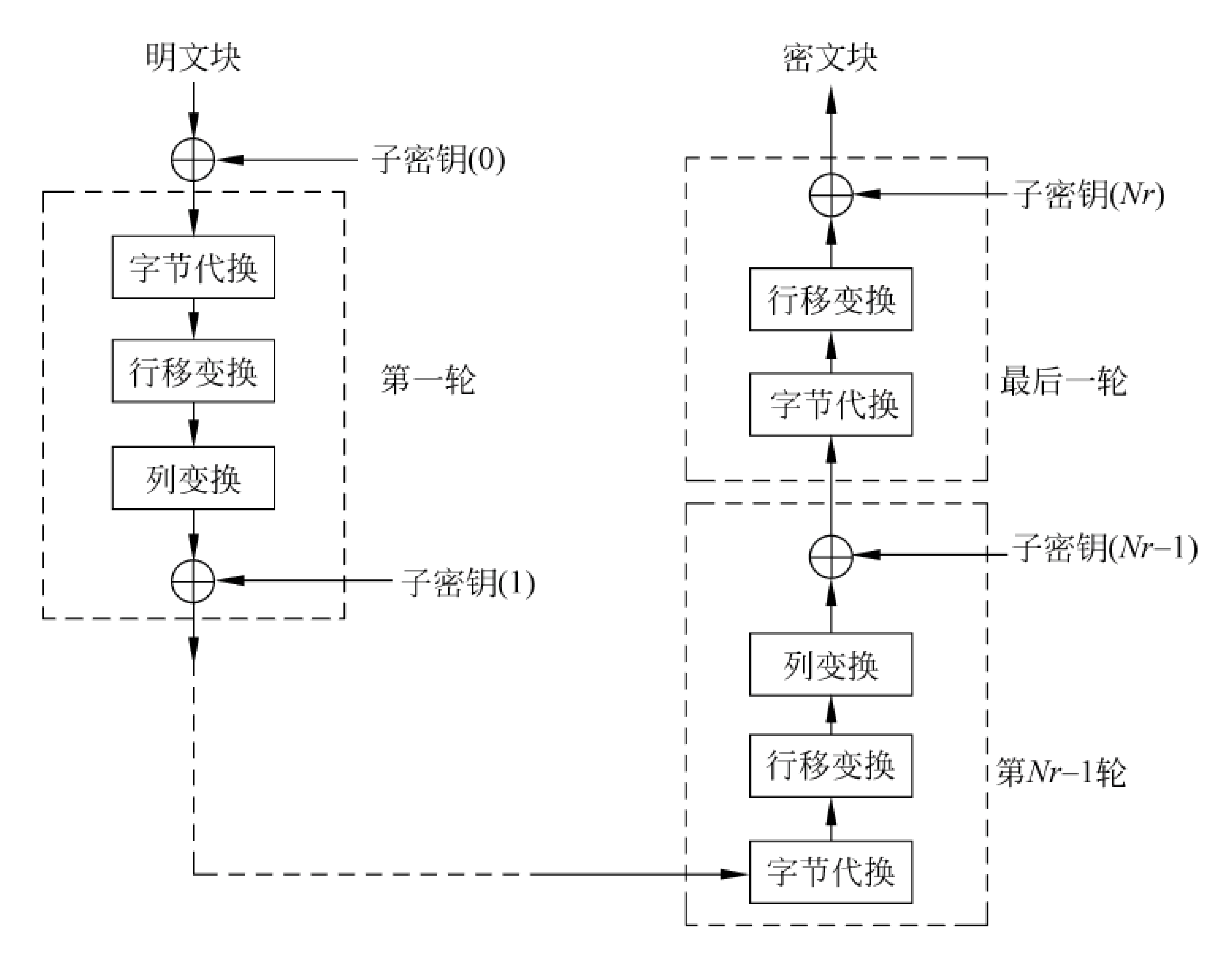


图 4-15 加密流程

1. 字节代换

字节代换(ByteSub)是非线形变换,独立地对状态的每个字节进行。代换表(即 *S* 盒,如表 4-16 所示)是可逆的,由以下两个变换的合成得到。

首先,将字节看作 GF(28)上的元素,映射到自己的乘法逆元,00 映射到自己。

表 4-16 *S* 盒

		Y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
X	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	Ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	Dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

其次，对字节做 GF(2^8)上的可逆的仿射变换。

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

字节代换的示意图如图 4-16 所示。

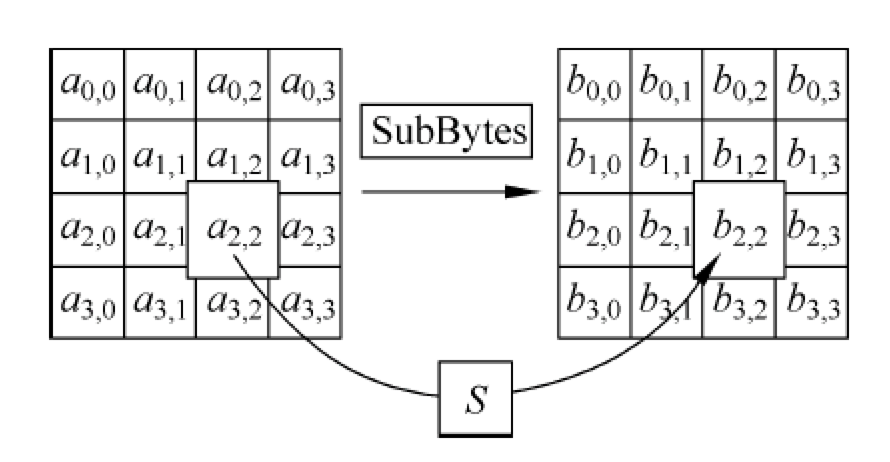


图 4-16　字节代换示意图

2. 行移位

行移位(ShiftRow)是将状态阵列的各行进行循环移位，不同状态行的位移量不同。第 0 行不移动，第 1 行循环左移 C_1 个字节，第 2 行循环左移 C_2 个字节，第 3 行循环左移 C_3 个字节。位移量 C_1、C_2、C_3 的取值与 N_b 有关。

按指定的位移量对状态的行进行的行移位运算记为 ShiftRow。

图 4-17 是行移位示意图。

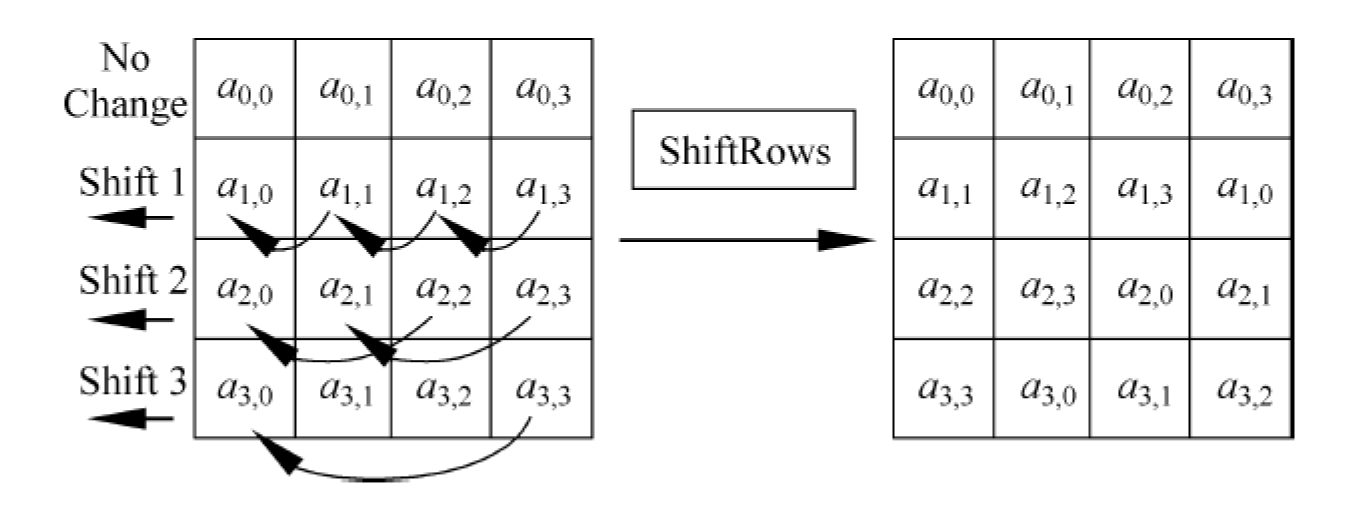


图 4-17　行移位示意图

3. 列混合

在列混合(MixColumn)变换中，将状态阵列的每个列视为 GF(2^8)上的多项式，再与一个固定的多项式 $c(x)$进行模 x^4+1 乘法。当然要求 $c(x)$是模 x^4+1 可逆的多项式，否则列混合变换就是不可逆的，因而会使不同的输入分组对应的输出分组可能相同。图 4-18 是列混合示意图。

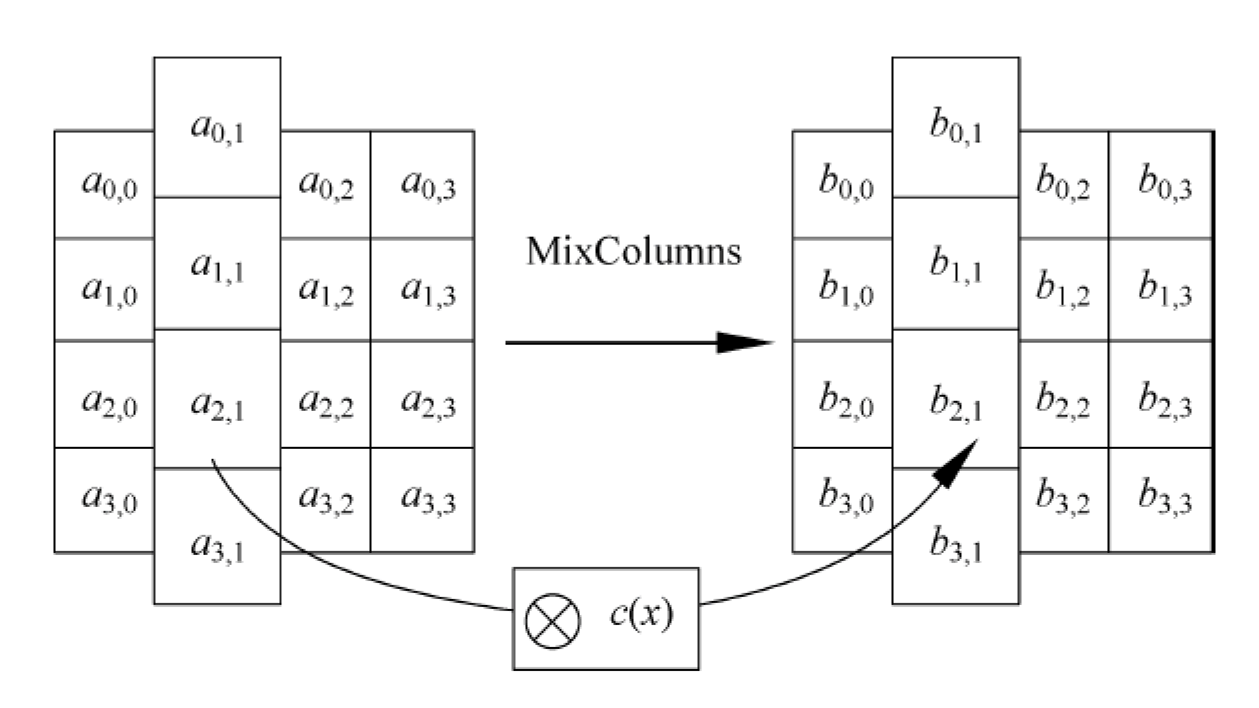


图 4-18 列混合示意图

4. 密钥加

密钥加(AddRoundKey)是将轮密钥简单地与状态进行逐比特异或。轮密钥由种子密钥通过密钥编排算法得到,轮密钥长度等于分组长度 N_b。图 4-19 是密钥加运算示意图。

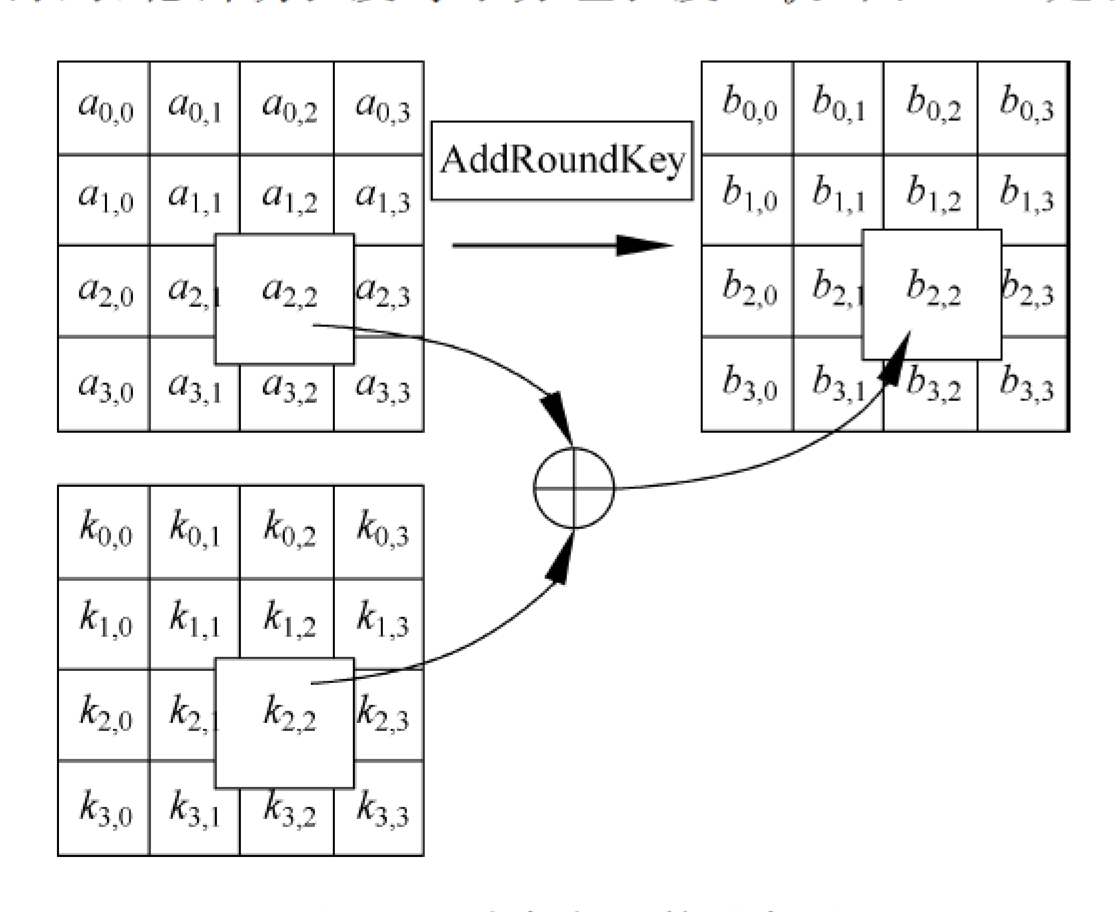


图 4-19 密钥加运算示意图

状态 State 与轮密钥 RoundKey 的密钥加运算表示为

```
AddRoundKey (State,RoundKey)
```

4.6.4 AES 算法的密钥编排

密钥编排指从种子密钥得到轮密钥的过程,它由密钥扩展和轮密钥选取两部分组成,其基本原则如下:

轮密钥的比特数等于分组长度乘以轮数加 1;

种子密钥被扩展成为扩展密钥;

轮密钥从扩展密钥中取,其中第 1 轮密钥取扩展密钥的前 N_b 个字,第 2 轮密钥取接下来的 N_b 个字,如此下去。

1. 密钥扩展

扩展密钥是以4字节字为元素的一维阵列，表示为 $W[N_b*(N_r+1)]$，其中前 N_k 个字取为种子密钥，以后每个字按递归方式定义。扩展算法根据 $N_k \leqslant 6$ 和 $N_k > 6$ 有所不同。

当 $N_k \leqslant 6$ 时，扩展算法如下：

```
KeyExpansion (byteKey[4 * Nk] ,W[Nb * (Nr + 1)])
{
    for (i = 0; i < Nk; i ++)
        W[i] = (Key[4 * i],Key[4 * i + 1],Key[4 * i + 2],Key[4 * i + 3] );
    for (i = Nk; i < Nb * (Nr + 1); i ++)
        {
            temp = W[i - 1];
            if (i % Nk = = 0)
            temp = SubByte (RotByte (temp))^Rcon[i /Nk];
            W[i] = W[i - Nk]^ temp;
        }
}
```

其中 $Key[4*N_k]$ 为种子密钥，看作以字节为元素的一维阵列。函数 SubByte ()返回4字节字，其中每一个字节都是用 Rijndael 的 S 盒作用到输入字对应的字节得到的。函数 RotByte()也返回4字节字，该字由输入的字循环移位得到，即当输入字为(a,b,c,d)时，输出字为(b,c,d,a)。

当 $N_k > 6$ 时，扩展算法如下：

```
KeyExpansion (byte Key[4 * Nk] ,W[Nb * (Nr + 1)])
{
    for (i = 0; i < Nk; i ++)
    W[i] = (Key[4 * i],Key[4 * i + 1],Key[4 * i + 2],Key[4 * i + 3] );
    for (i = Nk; i < Nb * (Nr + 1); i ++)
        {
         temp = W[i - 1];
         if (i % Nk = = 0)
         temp = SubByte (RotByte (temp))^Rcon[i /Nk];
         else if (i % Nk == 4)
         temp = SubByte (temp);
         W[i] = W[i - Nk]^ temp;
        }
}
```

$N_k > 6$ 与 $N_k \leqslant 6$ 的密钥扩展算法的区别在于：当 i 为 N_k 的4的倍数时，须先将前一个字 $W[i-1]$ 经过 SubByte 变换。

以上两个算法中，$Rcon[i/N_k]$ 为轮常数，其值与 N_k 无关。

2. 轮密钥选取

轮密钥 i(即第 i 个轮密钥)由轮密钥缓冲字 $W[N_b*i]$～$W[N_b*(i+1)]$ 给出，如图4-20所示。

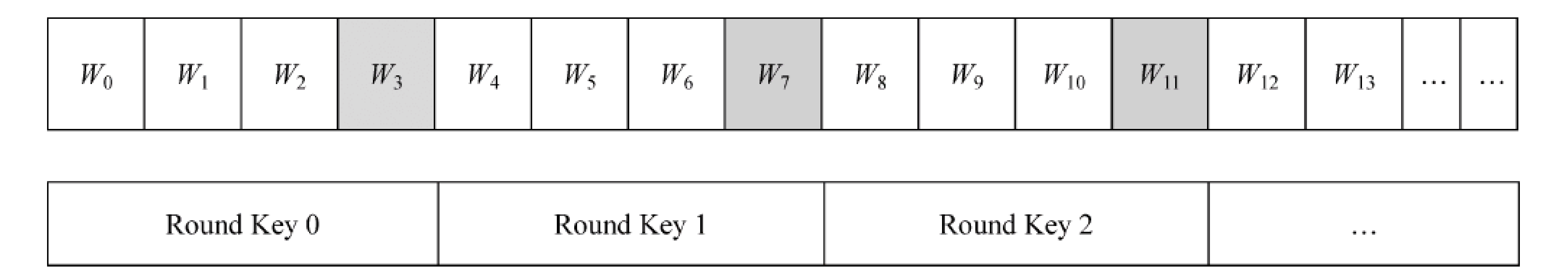


图 4-20 轮密钥的选取

4.7 AES 密码分析

针对 AES 的比较著名的攻击是旁道攻击。旁道攻击不攻击密码本身，而是攻击那些实现于不安全系统(会在不经意间泄漏信息)上的加密系统。美国国家安全局审核了所有参与竞选 AES 的最终入围者(包括 Rijndael)，认为它们均能够满足美国政府传递非机密文件的安全需要。2003 年 6 月，美国政府宣布 AES 可以用于加密机密文件：

AES 加密算法(使用 128、192 和 256 位密钥的版本)的安全性，在设计结构及密钥的长度上俱已到达保护机密信息的标准。最高机密信息的传递，则至少需要 192 或 256 位的密钥长度。用以传递国家安全信息的 AES 实现产品，必须先由国家安全局审核认证，方能被发放使用。

这标志着，由美国国家安全局 NSA 批准在最高机密信息上使用的加密系统首次可以被公开使用。许多大众化产品只使用 128 位密钥当作默认值；由于最高机密文件的加密系统必须保证数十年的安全性，故推测 NSA 可能认为 128 位太短，才以更长的密钥长度为最高机密的加密保留了安全空间。

通常破解一个区块加密系统最常见的方式，是先对其较弱版本(加密循环次数较少)尝试各种攻击。AES 中 128 位密钥版本有 10 个加密循环，192 位密钥版本有 12 个加密循环，256 位密钥版本则有 14 个加密循环。至 2006 年为止，最著名的攻击是针对 AES 7 次加密循环的 128 位密钥版本，8 次加密循环的 192 位密钥版本，和 9 次加密循环的 256 位密钥版本所作的攻击。

由于已遭破解的弱版的 AES，其加密循环数和原本的加密循环数相差无几，有些密码学家开始担心 AES 的安全性：要是有人能将该著名的攻击加以改进，这个区块加密系统就会被破解。在密码学的意义上，只要存在一个方法，比暴力搜索密钥还要更有效率，就能被视为一种“破解”。故一个针对 AES 128 位密钥的攻击若“只”需要 2120 计算复杂度(少于暴力搜索法 2128)，128 位密钥的 AES 就算被破解了；即便该方法在目前还不实用。从应用的角度来看，这种程度的破解依然太不切实际。最著名的暴力攻击法是 distributed. net 针对 64 位密钥 RC5 所做的攻击(该攻击在 2002 年完成。根据摩尔定律，到 2005 年 12 月，同样的攻击应该可以破解 66 位密钥的 RC5)。

其他的争议则着重于 AES 的数学结构。不像其他区块加密系统，AES 具有相当井然有序的代数结构。虽然相关的代数攻击尚未出现，但有许多学者认为，把安全性建立于未经透彻研究过的结构上是有风险的。Ferguson、Schroeppel 和 Whiting 因此写道：“我们很担心 Rijndael [AES]算法应用在机密系统上的安全性。”

2002 年，Nicolas Courtois 和 Josef Pieprzyk 发表名为 XSL 攻击的理论性攻击，试图展示 AES 一个潜在的弱点。但几位密码学专家发现该攻击的数学分析有点问题，推测应是作者的计算有误。因此，这种攻击法是否对 AES 奏效，仍是未解之谜。就现阶段而言，XSL 攻击 AES 的效果不十分显著，故将之应用于实际情况的可能性并不高。

接下来讨论一下线性密码分析方法：现在假设有一个攻击者拥有大量的用同一个未知密钥加密的明密文对，对每一个明密文对，将所有有可能的候选密钥来对最后一轮解密密文，对每一个候选密钥，计算包含在线性关系式中相关状态比特的异或值，然后确定上述的线性关系是否成立，如果成立，就在对应特定候选密钥的计算器加 1。

4.7.1　*S* 盒的输入输出分析

字节替换操作使用一个 *S* 盒对 State 的每个字节都进行独立的替换。它用于将输入或中间态的每一个字节通过一个简单的查表操作将其映射为另一个字节。把输入字节的高 4 位作为 *S* 盒的行值，低 4 位作为列值，然后取出 *S* 盒中对应行和列的元素作为输出，如表 4-17 所示。

表 4-17　*S* 盒替换表

		Y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
	0	63	7c	77	7b	F2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
X	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	E1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

与 DES 的 *S* 盒相比较，AES 的 *S* 盒能进行代数上的定义，而不像 DES 的 *S* 盒那样“随机代换”。

S 盒的构造方式如下：

行 x 和列 y 的字节值初始化成十六进制的$\{xy\}$。

把 *S* 盒中的每个字节映射为在有限域 GF(2^8)中的逆；{00}不变。

把 *S* 盒中的每个字节转换为二进制表示的($b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0$)，然后进行仿射变换。

AES采用有限域GF(2^8)上的字节运算。

由于S盒是AES算法中唯一的非线性部分，所以对AES的S盒的输入输出进行彻底的分析，具体方法如下：

让S的输入为($a_0a_1a_2a_3a_4a_5a_6a_7$)，S盒对应的输出为($b_0b_1b_2b_3b_4b_5b_6b_7$)，然后穷举S盒可能的所有输入，即($a_0a_1a_2a_3a_4a_5a_6a_7$)=(00000000～11111111)，给出所有对应的输出，如表4-18所示(由于表过大，就不全部给出了)。

表4-18 S盒输入输出对应表

$a_0a_1a_2a_3a_4a_5a_6a_7$	$b_0b_1b_2b_3b_4b_5b_6b_7$
00000000	01100011
00110110	00000101
11111111	00010110

通过表4-18，穷举输入输出的每一位进行异或，计算其等于0或等于1的概率(取概率较大的)(由于结果过于庞大，这里只给出部分结果)。

$a_0\oplus b_1=1$，概率为P：0.51。

$a_0\oplus b_2=1$，概率为P：0.55。

$a_0\oplus b_3=1$，概率为P：0.53。

$a_0\oplus b_4=0$，概率为P：0.56。

$a_0\oplus b_5=0$，概率为P：0.56。

$a_6\oplus a_7\oplus b_0\oplus b_2\oplus b_5\oplus b_6\oplus b_7=1$，概率为$P$：0.53。

$a_6\oplus a_7\oplus b_0\oplus b_3\oplus b_4\oplus b_5\oplus b_6=1$，概率为$P$：0.52。

$a_6\oplus a_7\oplus b_0\oplus b_3\oplus b_4\oplus b_5\oplus b_7=0$，概率为$P$：0.55。

$a_6\oplus a_7\oplus b_0\oplus b_3\oplus b_4\oplus b_6\oplus b_7=0$，概率为$P$：0.51。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_4=0$，概率为$P$：0.53。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_5=1$，概率为$P$：0.55。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_6=0$，概率为$P$：0.55。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_7=0$，概率为$P$：0.55。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_3\oplus b_4=1$，概率为$P$：0.53。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_3\oplus b_5=1$，概率为$P$：0.55。

$a_1\oplus a_2\oplus a_3\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_3\oplus b_6=0$，概率为$P$：0.50。

$a_0\oplus a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_4\oplus b_5\oplus b_6\oplus b_7=0$，概率为$P$：0.51。

$a_0\oplus a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_3\oplus b_4\oplus b_5\oplus b_6\oplus b_7=0$，概率为$P$：0.51。

$a_0\oplus a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7\oplus b_0\oplus b_2\oplus b_3\oplus b_4\oplus b_5\oplus b_6\oplus b_7=1$，概率为$P$：0.54。

$a_0\oplus a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7\oplus b_1\oplus b_2\oplus b_3\oplus b_4\oplus b_5\oplus b_6\oplus b_7=0$，概率为$P$：0.53。

$a_0\oplus a_1\oplus a_2\oplus a_3\oplus a_4\oplus a_5\oplus a_6\oplus a_7\oplus b_0\oplus b_1\oplus b_2\oplus b_3\oplus b_4\oplus b_5\oplus b_6\oplus b_7=0$，概率为$P$：0.51。

通过上面的几步计算，得到了65 025个有关输入输出的概率方程，其所有的概率分布在0.50至0.56之间，其中概率为0.56的方程有3091个，而这些方程将在后面的分析中用到。

由于 128 位 AES 较为复杂，将先引入一个简化 16 位 AES 进行分析，验证。

假设简化 AES 的 S 盒对应的输入输出表，如表 4-19 所示。

表 4-19　16 位 AES 的输入输出对应表

$a_0a_1a_2a_3$	$b_0b_1b_2b_3$	$a_0a_1a_2a_3$	$b_0b_1b_2b_3$
0000	1001	1000	0110
0001	0100	1001	0010
0010	1010	1010	0000
0011	1011	1011	0011
0100	1101	1100	1100
0101	0001	1101	1110
0110	1000	1110	1111
0111	0101	1111	0111

可以提取出概率为 0.75 的方程：

$$a_0 \oplus b_2 = 0$$
$$a_3 \oplus b_0 = 1$$
$$a_0 \oplus a_1 \oplus b_0 = 1$$
$$a_2 \oplus a_3 \oplus b_3 = 1$$
$$a_2 \oplus b_2 \oplus b_3 = 1$$
$$a_1 \oplus b_1 = 0$$
$$a_2 \oplus b_1 \oplus b_3 = 1$$
$$a_0 \oplus a_1 \oplus a_2 \oplus a_3 \oplus b_1 = 0$$
$$a_1 \oplus b_0 \oplus b_3 = 0$$
$$a_1 \oplus a_2 \oplus b_0 \oplus b_1 = 1$$
$$a_0 \oplus a_1 \oplus b_1 \oplus b_2 \oplus b_3 = 1$$
$$a_0 \oplus b_0 \oplus b_1 = 1$$

4.7.2　AES 的扩展密钥分析

由于 AES 属于迭代变换，所以扩展密钥是必不可少的，通过生成器产生 N_r+1 个轮密钥，每个轮密钥由 N_b 个字组成，共有 $N_b(N_r+1)$ 个字 $W[i]$，$i=0,1,\cdots,N_b(N_r+1)-1$。加密过程中，需要 N_r+1 个轮密钥（即子密钥），需要 $4(N_r+1)$ 个 32 位字。

令 key[]和 W[]分别用于存储扩展前、后的密钥。SubWord()、RotWord()分别是 S 盒的置换和以字节为单位的循环位移。$R_{\mathrm{con}}[i]=(\mathrm{RC}[i],'00','00','00')$，$\mathrm{RC}[i]=2\cdot\mathrm{RC}[i-1](i>1)$。字节运算是多项式运算，因此可以用多项式表示为

$$\mathrm{RC}[i]=x\cdot\mathrm{RC}[i-1]=x^{i-1}\bmod(x^8+x^4+x^3+x+1)\quad(i>1)$$

AES 密钥扩展算法的输入是 4 个字（每个字 32 位，共 128 位）。输入密钥直接被复制到扩展密钥数组的前 4 个字中，得到 $w[0]$，$w[1]$，$w[2]$，$w[3]$；然后每次用 4 个字填充扩展密钥数组余下的部分。在扩展密钥数组中，$w[i]$的值取决于 $w[i-1]$和 $w[i-4]$($i\geqslant4$)。

对 w 数组中下标不为 4 的倍数的元素，只是简单地异或，其逻辑关系为

$$w[i]=w[i-1]\oplus w[i-4]\quad(i\text{ 不为 4 的倍数})$$

对 w 数组中下标为 4 的倍数的元素，采用以下的计算方法：

(1) RotWord()：将前一个字的 4 个字节循环左移一个字节，即将字 $(b_0 b_1 b_2 b_3)$ 变为 $(b_1 b_2 b_3\ b_0)$。

(2) SubWord()：基于 S 盒对输入字中的每个字节进行 S 替换。

(3) 将步骤(2)的结果再与轮常量 $R_{con}[i]$ 异或运算。

(4) 将步骤(3)的结果再与 $w[i-4]$ 进行异或运算，即

$$W[i] = \mathrm{SubWord}(\mathrm{RotWord}(w[i-1])) \oplus R_{con}[i/4] \oplus w[i-4] \quad (i \text{ 为 4 的倍数})$$

对于三轮 AES 加密，需要原始密钥和三轮扩展密钥，即 K_0, K_1, K_2, K_3；

$K_0 = W[0]W[1]W[2]W[3] = (k_0, \cdots, k_{31})(k_{32}, \cdots, k_{63})(k_{64}, \cdots, k_{95})(k_{96}, \cdots, k_{127})$(128 位原始密钥)

$K_1 = W[4]W[5]W[6]W[7] = (k_{128}, \cdots, k_{159})(k_{160}, \cdots, k_{191})(k_{192}, \cdots, k_{223})(k_{224}, \cdots, k_{255})$

$K_2 = W[8]W[9]W[10]W[11] = (k_{256}, \cdots, k_{287})(k_{288}, \cdots, k_{319})(k_{320}, \cdots, k_{351})(k_{352}, \cdots, k_{383})$

$K_3 = W[12]W[13]W[14]W[15] = (k_{384}, \cdots, k_{415})(k_{416}, \cdots, k_{447})(k_{448}, \cdots, k_{479})(k_{480}, \cdots, k_{511})$

$W[4] = W[0] \oplus \text{“01000000”} \oplus \text{S-box}(k_8 k_9 k_{10} k_{11} k_{12} k_{13} k_{14} k_{15}) \oplus \text{S-box}(k_{16} k_{17} k_{18} k_{19} k_{20} k_{21} k_{22} k_{23}) \oplus \text{S-box}(k_{24} k_{25} k_{26} k_{27} k_{28} k_{29} k_{30} k_{31}) \oplus \text{S-box}(k_0 k_1 k_2 k_3 k_4 k_5 k_6 k_7)$

$W[5] = W[4] \oplus W[1]$

$W[6] = W[5] \oplus W[2]$

$W[7] = W[6] \oplus W[3]$

$W[8] = W[4] \oplus \text{“02000000”} \oplus \text{S-box}(k_{232} k_{233} k_{234} k_{235} k_{236} k_{237} k_{238} k_{239}) \oplus \text{S-box}(k_{240} k_{241} k_{242} k_{243} k_{244} k_{245} k_{246} k_{247}) \oplus \text{S-box}(k_{248} k_{249} k_{250} k_{251} k_{252} k_{253} k_{254} k_{255}) \oplus \text{S-box}(k_{224} k_{225} k_{226} k_{227} k_{228} k_{229} k_{230} k_{231})$

$W[9] = W[8] \oplus W[5]$

$W[10] = W[9] \oplus W[6]$

$W[11] = W[10] \oplus W[7]$

$W[12] = W[8] \oplus \text{“04000000”} \oplus \text{S-box}(k_{360} k_{361} k_{362} k_{363} k_{364} k_{365} k_{366} k_{367}) \oplus \text{S-box}(k_{368} k_{369} k_{370} k_{371} k_{372} k_{373} k_{374} k_{375}) \oplus \text{S-box}(k_{376} k_{377} k_{378} k_{379} k_{380} k_{381} k_{382} k_{383}) \oplus \text{S-box}(k_{352} k_{353} k_{354} k_{355} k_{356} k_{357} k_{358} k_{359})$

令

$\text{S-box}(k_8 k_9 k_{10} k_{11} k_{12} k_{13} k_{14} k_{15}) = l_0 l_1 l_2 l_3 l_4 l_5 l_6 l_7$

$\text{S-box}(k_{16} k_{17} k_{18} k_{19} k_{20} k_{21} k_{22} k_{23}) = l_8 l_9 l_{10} l_{11} l_{12} l_{13} l_{14} l_{15}$

$\text{S-box}(k_{24} k_{25} k_{26} k_{27} k_{28} k_{29} k_{30} k_{31}) = l_{16} l_{17} l_{18} l_{19} l_{20} l_{21} l_{22} l_{23}$

$\text{S-box}(k_0 k_1 k_2 k_3 k_4 k_5 k_6 k_7) = l_{24} l_{25} l_{26} l_{27} l_{28} l_{29} l_{30} l_{31}$

$\text{S-box}(k_{232} k_{233} k_{234} k_{235} k_{236} k_{237} k_{238} k_{239}) = l_{32} l_{33} l_{34} l_{35} l_{36} l_{37} l_{38} l_{39}$

$\text{S-box}(k_{240} k_{241} k_{242} k_{243} k_{244} k_{245} k_{246} k_{247}) = l_{40} l_{41} l_{42} l_{43} l_{44} l_{45} l_{46} l_{47}$

$\text{S-box}(k_{248} k_{249} k_{250} k_{251} k_{252} k_{253} k_{254} k_{255}) = l_{48} l_{49} l_{50} l_{51} l_{52} l_{53} l_{54} l_{55}$

$\text{S-box}(k_{224} k_{225} k_{226} k_{227} k_{228} k_{229} k_{230} k_{231}) = l_{56} l_{57} l_{58} l_{59} l_{60} l_{61} l_{62} l_{63}$

$\text{S-box}(k_{360} k_{361} k_{362} k_{363} k_{364} k_{365} k_{366} k_{367}) = l_{64} l_{65} l_{66} l_{67} l_{68} l_{69} l_{70} l_{71}$

$\text{S-box}(k_{368} k_{369} k_{370} k_{371} k_{372} k_{373} k_{374} k_{375}) = l_{72} l_{73} l_{74} l_{75} l_{76} l_{77} l_{78} l_{79}$

$\text{S-box}(k_{376} k_{377} k_{378} k_{379} k_{380} k_{381} k_{382} k_{383}) = l_{80} l_{81} l_{82} l_{83} l_{84} l_{85} l_{86} l_{87}$

S-box$(k_{352}k_{353}k_{354}k_{355}k_{356}k_{357}k_{358}k_{359})=l_{88}l_{89}l_{90}l_{91}l_{92}l_{93}l_{94}l_{95}$

由此,可以推导出扩展密钥和原始密钥的关系是

$k_i=k_{i-128}\oplus l_{i-128}\quad(128\leqslant i<159)$

$k_{159}=k_{31}\oplus l_{31}\oplus 1$

$k_i=k_{i-32}\oplus k_{i-128}\quad(160\leqslant i<255)(288\leqslant i\leqslant 383)(416\leqslant i\leqslant 511)$

$k_i=k_{i-128}\oplus l_{i-224}\quad(416\leqslant i\leqslant 511,i\neq 286)$

$k_{286}=k_{158}\oplus l_{63}\oplus 1$

$k_i=k_{i-128}\oplus l_{i-320}\quad(384\leqslant i\leqslant 415,i\neq 413)$

$k_{413}=k_{285}\oplus l_{93}\oplus 1$

对于简化 16 位 AES,将得到

$K_0=W[0]W[1]=(k_0,\cdots,k_7)(k_8,\cdots,k_{15})$

$K_1=W[2]W[3]=(k_{16},\cdots,k_{23})(k_{24},\cdots,k_{31})$

$K_2=W[4]W[5]=(k_{32},\cdots,k_{39})(k_{40},\cdots,k_{47})$

$W[2]=W[0]\oplus$'10000000'$\oplus$S-box$(k_{12}k_{13}k_{14}k_{15})\oplus$S-box$(k_8k_9k_{10}k_{11})$

$W[3]=W[1]\oplus W[2]$

$W[4]=W[2]\oplus$'00110000'$\oplus$S-box$(k_{28}k_{29}k_{30}k_{31})\oplus$S-box$(k_{24}k_{25}k_{26}k_{27})$

$W[5]=W[3]\oplus W[4]$

令

S-box$(k_{12}k_{13}k_{14}k_{15})=l_0l_1l_2l_3$

S-box$(k_8k_9k_{10}k_{11})=l_4l_5l_6l_7$

S-box$(k_{28}k_{29}k_{30}k_{31})=l_8l_9l_{10}l_{11}$

S-box$(k_{24}k_{25}k_{26}k_{27})=l_{12}l_{13}l_{14}l_{15}$

由此可以推出

$k_{16}=k_0\oplus l_0\oplus 1,k_i=k_{i-16}\oplus k_{i-16}\quad(17\leqslant i<23)$

$k_{34}=k_{18}\oplus l_{10}\oplus 1,k_{35}=k_{19}\oplus l_{11}\oplus 1$

$k_i=k_{i-16}\oplus k_{i-24}\quad(i=32,33,36,37,38,39)$

$k_i=k_{i-8}\oplus k_{i-16}\quad(15\leqslant i<31)(40\leqslant i<47)$

4.7.3 AES 线性密码分析

简化 AES 加密过程是在一个 2×2 的字节矩阵上运作,有一个 16 位的原始密钥,记为 $k_0k_1\cdots k_{15}$。这密钥需要扩展到 48 位 $k_0k_1\cdots K_{47}$,其中前 16 位是原始密钥而其他的是根据密钥扩展算法而扩展得来的子密钥。各轮 AES 加密循环(除最后一轮外)包含 4 个步骤(和 AES 类同):

A(轮密钥加密)——矩阵中的每一个字节都与该轮密钥做异或运算,如图 4-21 所示。

N_0	N_2
N_1	N_3

$\oplus$

$W[1]$	$W[1+1]$

$=$

N'_0	N'_2
N'_1	N'_3

图 4-21 16 位 AES 轮密钥加密

NS(S 盒替换)——通过一个非线性的替换函数,用查找表的方式把每个字节替换成对应的字节,具体操作如图 4-22 所示。

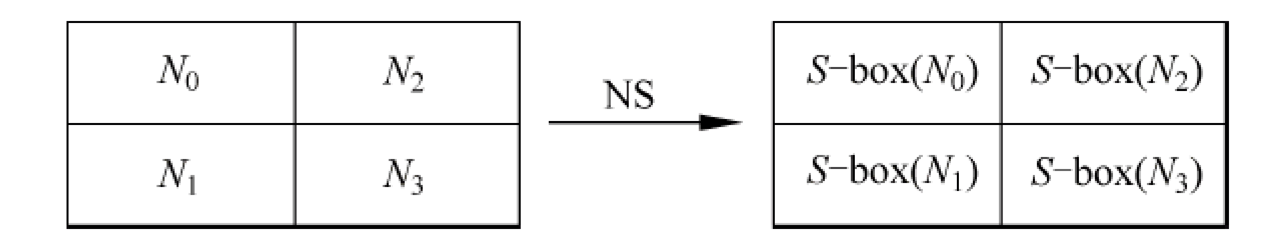


图 4-22　16 位 AES 的 S 盒替换

SR(行位移)——将矩阵中的每个横列进行循环式移位,具体操作如图 4-23 所示。

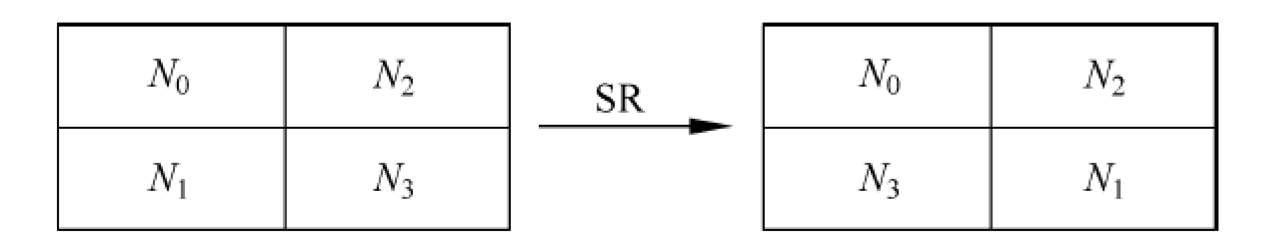


图 4-23　16 位 AES 行位移

MC(列混淆)——为了充分混合矩阵中各个直行的操作。MC 是 MixColumn 运算的缩写。$[N_i, N_j]$被认为是 GF(16)$[z]/(z^2+1)$的元素 $N_j z+N_j$。函数 MC 乘以多项式 $c(z)=x^2 z+1$ 的每列。

最后一个加密循环中省略 MixColumns 步骤,而以另一个 AddRoundKey 取代。

函数 RotNib 被定义为 RotNib$(N_0 N_1)=N_1 N_0$ 而函数 SubNib 被定义为 SubNib$(N_0 N_1)=S\text{-box}(N_0)S\text{-box}(N_1)$。这两个函数名分别是行位移,替代变换的缩写。

简化 AES 算法是通过使用扩展密钥 $k_0 k_1 \cdots k_{47}$ 对 16 位明文加密产生 16 位密文。假设 $p_0 p_1 \cdots p_{15}$ 是明文,$c_0 c_1 \cdots c_{15}$ 是密文,加密算法由 8 种构造函数加密明文。所以,

$$c_0 c_1 \cdots c_{15} = A_{K_2} \oplus \text{SR} \oplus \text{NS} \oplus A_{K_1} \oplus \text{MC} \oplus \text{SR} \oplus \text{NS} \oplus A_{K_0}(p_0 p_1 \cdots p_{15})$$

其中 $A_{K_i}(p)=K_i \oplus p$

在简化 AES 加密中构造函数 $A_{K_i} \oplus \text{MC} \oplus \text{SR} \oplus \text{NS}$ 被认为是用在第 i 轮加密的。简化到两轮,A_{K_0} 优先用于第一轮,MC 忽略第二轮。

分组算法很多,其中较经典的包括 DES、3DES、AES 和 IDEA,表 4-20 将对这些算法进行比较分析。

表 4-20　DES、3DES、AES、IDEA 算法比较

类型	定义	密钥长度	分组长度	循环次数	安全性
DES	数据加密标准,速度较快,适用于加密大量数据的场合	56	64	16	依赖密钥受穷举搜索法攻击
3DES	是基于 DES 的对称算法,对一块数据用三个不同的密钥进行三次加密,强度更高	112　168	64	48	军事级,可抗差值分析和相关分析
AES	高级加密标准,对称算法,是下一代的加密算法标准,速度快,安全级别高,目前 AES 标准的一个实现是 Rijndael 算法	128 192 256	64	10 12 14	安全级别高,高级加密标准
IDEA	国际数据加密算法,使用 128 位密钥提供非常强的安全性	128	64	8	能抵抗差分密码分析的攻击

1. DES

算法的入口参数有三个：Key、Data、Mode。其中 Key 为 8 个字节共 64 位，是 DES 算法的工作密钥。

Data 也为 8 个字节 64 位，是要被加密或被解密的数据；Mode 为 DES 的工作方式，有两种：加密或解密。

如 Mode 为加密，则用 Key 去把数据 Data 进行加密，生成 Data 的密码形式(64 位)作为 DES 的输出结果；如 Mode 为解密，则用 Key 去把密码形式的数据 Data 解密，还原为 Data 的明码形式(64 位)作为 DES 的输出结果。

在通信网络的两端，双方约定一致的 Key，在通信的源点用 Key 对核心数据进行 DES 加密，然后以密码形式在公共通信网(如电话网)中传输到通信网络的终点，数据到达目的地后，用同样的 Key 对密码数据进行解密，便再现了明码形式的核心数据。这样，便保证了核心数据(如 PIN、MAC 等)在公共通信网中传输的安全性和可靠性。通过定期在通信网络的源端和目的端同时改用新的 Key，便能更进一步提高数据的保密性，这正是现在金融交易网络的流行做法。

2. 3DES

3DES 是 DES 加密算法的一种模式，它使用 3 条 64 位的密钥对数据进行三次加密。数据加密标准(DES)是美国的一种由来已久的加密标准，它使用对称密钥加密法，是 DES 向 AES 过渡的加密算法(1999 年，NIST 将 3DES 指定为过渡的加密标准)，是 DES 的一个更安全的变形。它以 DES 为基本模块，通过组合分组方法设计出分组加密算法。

设 $E_K()$ 和 $D_K()$ 代表 DES 算法的加密和解密过程，K 代表 DES 算法使用的密钥，P 代表明文，C 代表密表，这样，

3DES 加密过程为 $C=E_{K_3}(D_{K_2}(E_{K_1}(P)))$。

3DES 解密过程为 $P=D_{K_1}(E_{K_2}(D_{K_3}(C)))$。

K_1、K_2、K_3 决定了算法的安全性，若三个密钥互不相同，本质上就相当于用一个长为 168 位的密钥进行加密。多年来，它在对付强力攻击时是比较安全的。若数据对安全性要求不那么高，K_1 可以等于 K_3。在这种情况下，可以表示为

$E_{K_1}(D_{K_2}(E_{K_3}(P)))=C$　加密

$D_{K_1}(E_{K_2}(D_{K_3}(C)))=P$　解密密钥的有效长度为 112 位

3. AES

AES 也是分块对数据加密的，只是块的长度并不像 DES 那样定为 64 位。Rijndael 的密钥长度可以从 128 位起以 32 位为间隔递增到 256 位。Rijndael 算法完全公开、安全性好、运算速度极快。如果取密钥长度为 128 位，想用穷举法破解密钥，就算有一台内含 1000 亿个处理器的计算机，并且每个处理器每秒处理 100 亿个密钥，也要运行 100 亿年才能搜索完整个密钥空间。

4. IDEA

算法安全性比 DES 好(和 AES 差不多)，能抵抗差分密码分析的攻击，而 DES 不行。

IDEA 的加密速度比 DES 快，加密数据速率可达到 177MB/s，也和 AES 差不多。

习题

1. 简述 Feistel 密码的重要性。
2. 混淆和扩散的差别是什么？
3. 在 S-DES 的密钥生成过程中，初始置换 $P10$ 和两个 LS-1 置换的作用是什么？
4. 利用 S-DES 密码体制加密明文 star，主密钥 $K=1010010111$。
5. 利用 S-DES 密码体制解密密文 star，主密钥 $K=1010010111$。

第5章 公钥密码技术

在公钥密码体制以前的整个密码学史中，所有的密码算法，包括原始手工计算的、由机械设备实现的以及由计算机实现的，都是基于代替和置换这两个基本工具的。而公钥密码体制则为密码学的发展提供了新的理论和技术基础，一方面公钥密码算法的基本工具不再代换和置换，而是数学函数；另一方面公钥密码算法以非对称的形式使用两个密钥，两个密钥的使用对保密性、密钥分配、认证等都有着深刻的意义。可以说公钥密码体制的出现在密码学史上是一个最大的而且是唯一真正的革命。

5.1 概述

5.1.1 公钥密码体制的提出

对称密码体制(分组加密体制)通过对明文的分组可以进行快速的加密，但其仍存在缺陷。

1. 密钥分配问题

通信双方要进行加密通信，需要通过秘密的安全信道协商加密密钥，而这种安全信道可能很难实现。

很久以来，“密钥分发”问题一直困扰着密码专家。比如，第二次世界大战时，德国高级指挥部每个月都需要分发《每日密钥》月刊给所有 Enigma 机的操作员。而且，即使 u 型潜艇大多数时间都远离基地，它也不得不想办法获得最新的密钥。美国政府的密钥是 COMSEC(通信安全局的缩写)掌管和分发的。20 世纪 70 年代，COMSEC 每天分发的密钥数以吨计。当装载着 COMSEC 密钥的船靠港时，密码分发员会到甲板上收集各种卡片、纸带以及软盘和其他一切储存密钥的介质。然后，把它们分发给客户。

2. 密钥管理问题

在有多个用户的网络中，任何两个用户之间都需要有共享的密钥，当网络中的用户 n 很大时，需要管理的密钥数目也非常大 $c(n,2)=n(n-1)/2$。若 $n=1000$，则 $C(1000,2)\approx$ 500 000，庞大的密钥如何管理？如何定期更换？这些都是一个十分复杂的工程。更有甚者，每一个用户与其他 $n-1$ 个用户的保密通信，需保存 $n-1$ 个密钥，如果将其记在本子上

或存储在计算机内部都是十分不安全的。

3. 没有签名功能

当主体 A 收到主体 B 的电子文档(电子数据)时,无法向第三方证明此电子文档确实来源于 B。

5.1.2 公钥密码体制的原理

公钥密码算法最大的特点是采用两个相关密钥将加密和解密分开,其中一个密钥是公开的,称为公开密钥,简称公钥,用于加密;另一个密钥是用户专用的,因而是保密的,称为秘密密钥,简称密钥,用于解密。因此公钥密码体制也称为双钥密码体制。算法有以下重要特性:已知密码算法和加密密钥,求解密密钥在计算上是不可行的。

公钥密码体制的加密解密原理如图 5-1 所示。

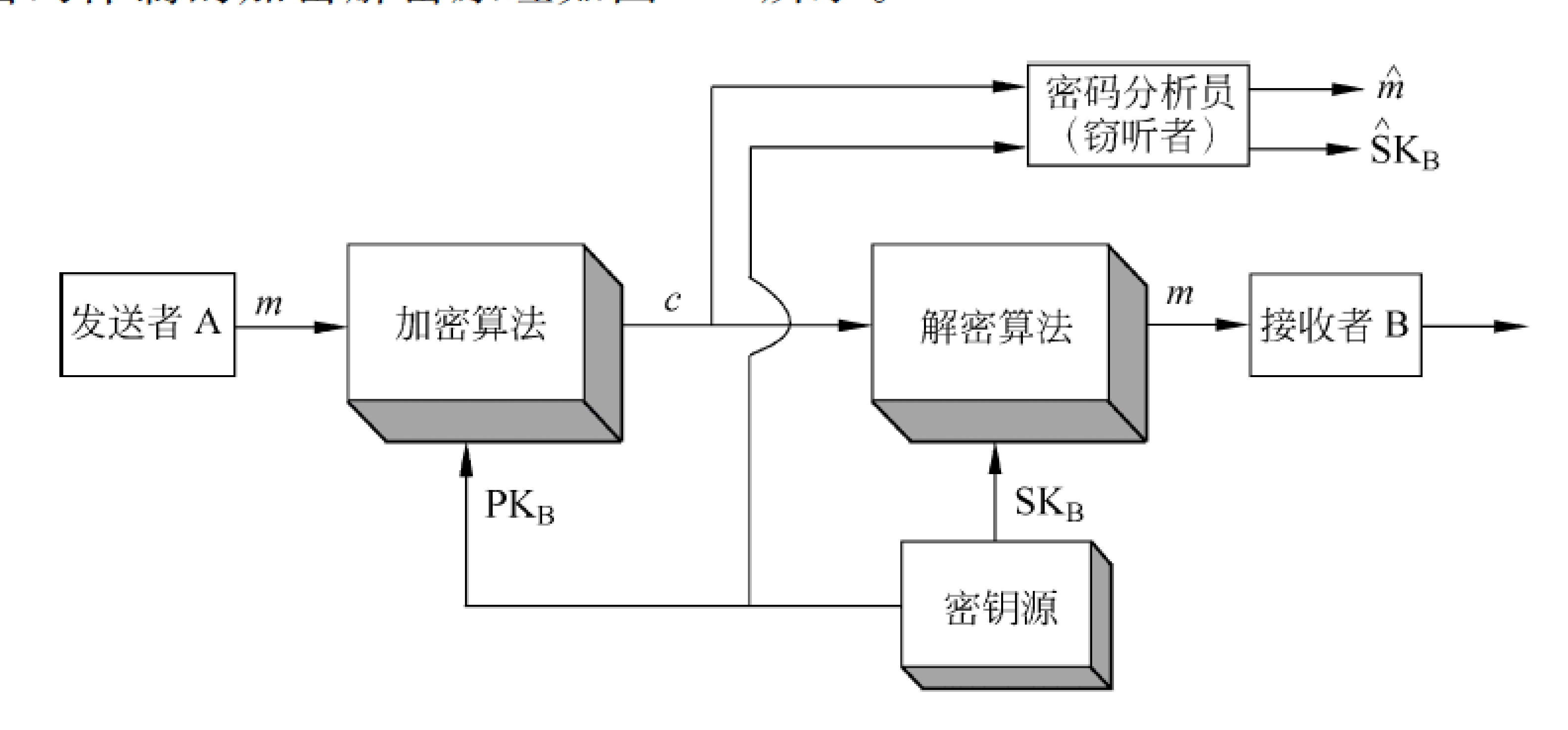


图 5-1 公钥密码体制的加解密原理图

整个加解密过程可以分成以下步骤:

(1) 要求接收消息的端系统,产生一对用来加密和解密的密钥,如图中的接收者 B,产生一对密钥 PK_B,SK_B,其中 PK_B 是公钥,SK_B 是私钥。

(2) 端系统 B 将加密密钥(如图中的 PK_B)予以公开,另一密钥则被保密(图中的 SK_B)。

(3) A 要想向 B 发送消息 m,则使用 B 的公钥加密 m。

(4) B 收到密文后,用自己的密钥 SK_B 解密。

因为只有 B 知道 SK_B,所以其他人都无法对密文解密。

公钥加密算法不仅能用于加解密,还能用于对发方 A 发送的消息 m 提供认证,如图 5-2 所示。用户 A 用自己的密钥 SK_A 对 m 加密,将密文发往 B。B 用 A 的公钥 PK_A 解密。

因为从 m 得到的密文是经过 A 的密钥 SK_A 加密的,只有 A 才能做到。因此密文可当作 A 对 m 的数字签字。另一方面,任何人只要得不到 A 的密钥 SK_A 就不能篡改 m,所以以上过程获得了对消息来源和消息完整性的认证。

在实际应用中,特别是用户数目很多时,以上认证方法需要很大的存储空间,因为每个文件都必须以明文形式存储以方便实际使用,同时还必须存储每个文件被加密后的密文形式即数字签字,以便在有争议时用来认证文件的来源和内容。改进的方法是减小文件的数字签字的大小,即先将文件经过一个函数压缩成长度较小的比特串,得到的比特串称为认证

符。认证符具有这样一个性质：如果保持认证符的值不变而修改文件，这在计算上是不可行的。用发送者的密钥对认证符加密，加密后的结果为原文件的数字签字。这一内容将在后续章节详细介绍。

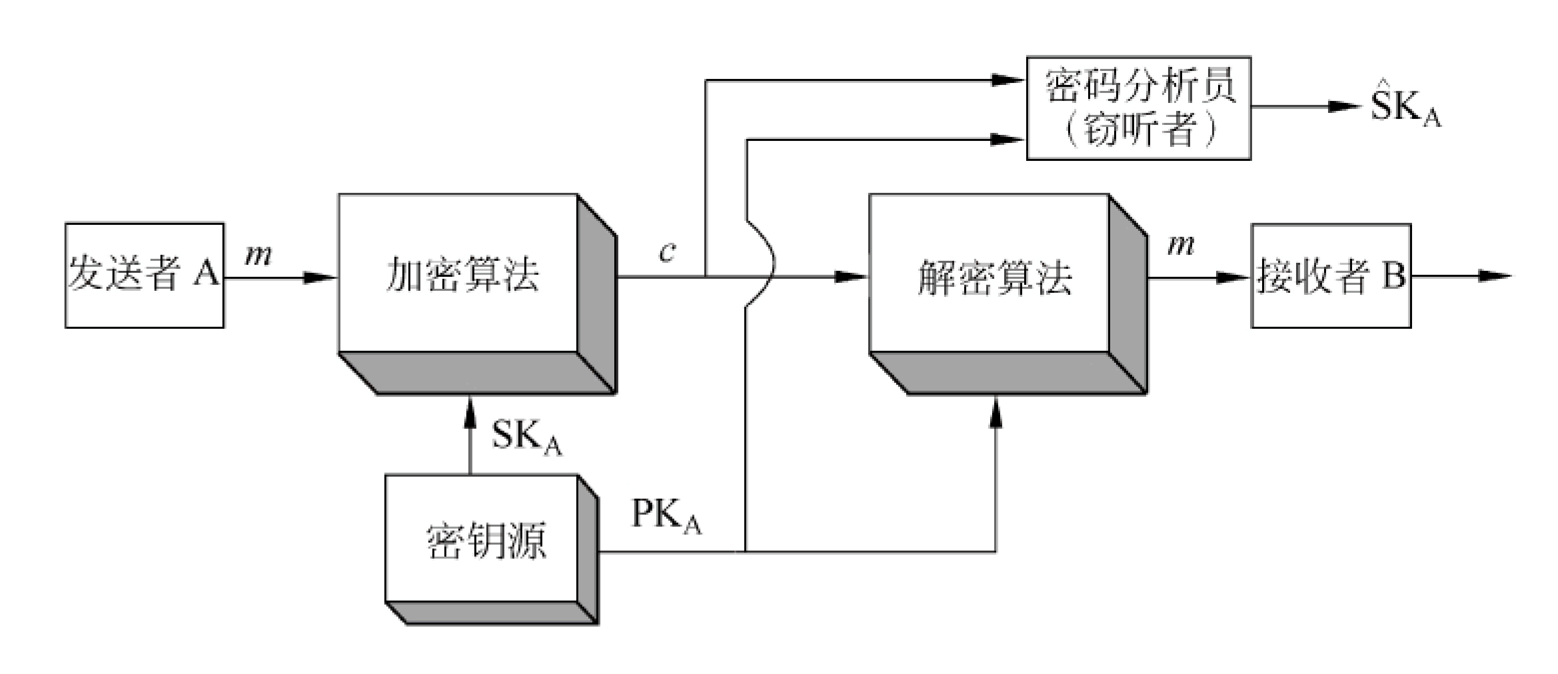


图 5-2 公钥密码体制的认证原理图

以上认证过程中，由于消息是由用户自己的密钥加密的，所以消息不能被他人篡改，但却能被他人窃听。这是因为任何人都能用用户的公钥对消息解密。为了同时提供认证功能和保密性，可使用双重加解密。

5.1.3 Diffie-Hellman 密钥交换算法

在 20 世纪 70 年代中期，斯坦福大学的研究生 Whirefield Diffie 和教授 Martin Hellman 特别研究了密钥分发问题。二人提出了一个方案，由此能够通过交换公开信息建立一个共享的秘密，他们可以在公开的信道上通信，以偷听者可读的形式来回传送信息，同时生成一个不公开的秘密数值，然后通信双方能够使用这个秘密数值作为对称会话密钥，这个方案称为 Diffie-Hellman，或者 DH。DH 解决了一个密钥共享问题。它的安全性是基于计算离散对数的困难性的。

这里涉及两个概念，一个是第 3 章中的本原元，一个是离散对数。

定义 5-1 设 p 是素数，a 是 p 的本原根，即 $a^1, a^2, \cdots\cdots, a^{p-1}$ 在 mod p 下产生 1 到 $p-1$ 的所有值，所以对 $b \in \{1, \cdots\cdots, p-1\}$，有唯一的 $i \in \{1, \cdots\cdots, p-1\}$ 使得 $b \equiv a^i \bmod p$。称 i 为模 p 下以 a 为底 b 的离散对数，记为 $i \equiv \log_a b(\bmod p)$。

当 a、p、i 已知时，可以比较容易地求出 b，但如果已知 a、b 和 p，求 i 则非常困难。

如果两个通信主体 Alice 和 Bob 希望在公开信道上建立密钥，则利用 Diffie-Hellman 密钥交换算法的建立过程为

(1) 选择一个大素数 p（200 位左右）。

(2) 计算 p 的一个本原元 a。

(3) Alice 选择一个密钥(Secret Key(number) $X_A < p$)。

(4) Bob 选择一个密钥(Secret Key(number) $X_B < p$)。

(5) Alice 和 Bob 计算他们的公开密钥。

$Y_A = a^{X_A} \bmod p$ 以及 $Y_B = x^{X_B} \bmod p$。

(6) Alice 和分别公开 Y_A，Y_B。

Diffie-Hellman 密钥交换算法的密钥交换过程为

（1）Alice 计算共享密钥 $K_1=X_B^{X_A} \bmod p$。

（2）Bob 计算共享密钥 $K_2=Y_A^{X_B} \bmod p$。

（3）计算：$K_1 =Y_B^{X_A} \bmod p=(a^{X_B} \bmod p)^{X_A} \bmod p$

$=a^{X_B X_A} \bmod p=(a^{X_A})^{X_B} \bmod p$

$=(a^{X_A} \bmod p)^{X_B} \bmod p=Y_A^{X_B} \bmod p=K_2$

所以，可以共享密钥。

这里的安全性主要体现在：如果只知道 a、p、Y_A、Y_B，想计算 X_A 和 X_B 是很困难的。

例如：选取素数 $p=97$，及本原元 $a=5$，Alice 选取密钥 $X_A=36$，Bob 选取密钥 $X_B=58$。

计算 Alice 的公钥 $Y_A=5^{36} \bmod 97=50$；Bob 的公钥 $Y_B=5^{58} \bmod 97=44$。

Alice 计算共享密钥 $K_1=44^{36} \bmod 97=75$；Bob 计算共享密钥 $K_2=50^{58} \bmod 97=75$，从而实现了密钥共享。

5.2 RSA 概述

RSA 为目前最著名的公开密钥密码系统，是由三位麻省理工学院（MIT）的学者罗纳德·李维斯特（Ron Rivest）、阿迪·萨莫尔（Adi Shamir）和伦纳德·阿德曼（Leonard Adleman）于 1978 年提出的。RSA 就是他们三人姓氏开头字母拼在一起组成的。RSA 密码系统可用于加解密、数字签章、密钥交换等，其安全性是建立于因子分解的困难度的。因子分解问题是指给定一合成数 n 为两个大素数 p 与 q 的乘积，欲分解 n 在计算上不可行。

RSA 密码体制分成两个部分：密钥生成和加解密算法。

5.2.1 密钥生成

（1）选两个大素数 p 和 q。

（2）计算 $n=pq$，$\varphi(n)=(p-1)(q-1)$，其中 $\varphi(n)$是 n 的欧拉函数值。

（3）选一整数 e（公钥），满足 $1<e<\varphi(n)$，且 $\gcd(\varphi(n),e)=1$。

（4）计算私钥 d，满足 $de\equiv 1 \bmod \varphi(n)$ ，即 $d=e^{-1} \bmod \varphi(n)$。

可以公开 n 和 e，只要不知道 p 和 q，就很难计算 $\varphi(n)$，就不知道私钥 d。

例 5-1 设 $p=7$，$q=11$，$e=13$，求 n、$\varphi(n)$和 d。

解：$n=pq=77$

$\varphi(n)=(p-1)(q-1)=6\times 10=60$

判断：

$\gcd(\varphi(n),e)=\gcd(13,60)=1$ 成立

$d=e^{-1} \bmod \varphi(n)=13^{-1} \bmod 60$

用扩展欧几里得算法可得 $d=-23 \bmod 60=37$

Q	X_1	X_2	X_3	Y_1	Y_2	Y_3
	1	0	60	0	1	13
4	0	1	13	1	−4	8
1	1	−4	8	−1	5	5
1	−1	5	5	2	−9	3
1	2	−9	3	−3	14	2
1	−3	14	2	5	−23	1

5.2.2　加解密算法

(1) 加密：$m<n$ 为明文，计算密文 $c=m^e \bmod n$。

(2) 解密：计算明文 $m=c^d \bmod n$

$$c^d \bmod n=(m^e \bmod n)^d \bmod n = m^{ed} \bmod n = m^{1+k\times\varphi(n)} \bmod n(\text{因为 } ed \equiv 1 \bmod \varphi(n))$$
$$=(m\times m^{k\times\varphi(n)}) \bmod n = (m \bmod n \times m^{k\times\varphi(n)} \bmod n) \bmod n$$
$$= m \bmod n(\text{欧拉定理}) = m$$

例 5-2　设 $p=7$、$q=17$、$e=5$、明文 $m=19$，求 $\varphi(n)$、d 和密文 c。

解：$n=pq=119$

$\varphi(n)=(p-1)(q-1)=96$

判断：$\gcd(\varphi(n),e)=\gcd(96,5)=1$ 成立。

$d=e^{-1} \bmod \varphi(n)=5^{-1} \bmod 96$

用扩展欧几里得算法可得 $d=77$

加密：$c=m^e \bmod n=19^5 \bmod 119=66$

对密文 c 的解密 $m=c^d \bmod n=66^{77} \bmod 119=19$，这里涉及了大数模幂乘的计算。

5.2.3　大数模幂乘的计算

对于大数模幂乘 $a^m \bmod n$ 的计算，可以采用快速取模指数算法来实现，具体方法包括以下两种方法，其伪码表示为

(1) m 的二进制表示为 $b_k b_{k-1}\cdots\cdots b_0$，其中 $b_i=\{0,1\}(i=0,1,\cdots\cdots,k)$。

```
c = 0; d = 1;
For  i = k downto 0
     {
          c = 2 × c;
          d = (d × d) mod n;
          if bi = 1 then
               {
               c = c + 1;
               d = (d × a) mod n
               }
     }
return d
```

(2) 步骤如下。

① $b=m$；$c=a$；$d=1$。

② 如果 $b=0$,输出结果 d。

③ 如果 b 是奇数,转到⑤。

④ $b=b/2$;$c=c^2 \bmod n$,转到③。

⑤ $b=b-1$；$d=(c\times d) \bmod n$,转到②。

例 5-3 用两种方法计算 $30^{37} \bmod 77$。

解：第一种方法。

37 的二进制表示为$(100101)_2$。

i	5	4	3	2	1	0
b_i	1	0	0	1	0	1
c	1	2	4	9	18	37
d	30	53	37	29	71	2

第二种方法。

b	C	d
37	30	1
36	↓	30
18	53	↓
9	37	↓
8	↓	32
4	60	↓
2	58	↓
1	53	↓
0	↓	2

5.2.4 素数判断

密钥生成需要两个很大的素数,然而现在却没有一个特别有效的算法来判断一个大数是否为素数。Miller-Rabin 素数判定算法是其中一个相对较好的算法。

定义 5-2 设 $n>2$ 是一个奇数,设 $n-1=2^s m$,其中 s 是非负整数,$m>0$ 是奇数。设 $0<b<n$,如果

$$b^m \equiv 1(\bmod n)$$

或者存在一个 r,$0\leqslant r<s$,使得

$$b^{2^r m} \equiv -1(\bmod n)$$

则称 n 通过以 b 为基的 Miller-Rabin 测试。

定理 5-1 设 $p>2$ 是一个素数。对任意整数 $b>0$,如果 $\gcd(b,p)=1$,则 p 一定可以通过以 b 为基的 Miller-Rabin 测试。

证明：设 $p-1=2^s m$,其中 s 是非负整数,$m>0$ 是奇数。令

$$T_k = b^{\frac{(p-1)}{2^k}} \bmod p = b^{2^{s-k}m} \bmod p$$

$k=0,1,2,\cdots,s$。因为 p 是素数，$\gcd(b,p)=1$，所以由 Fermat 定理知

$$b^{p-1} \equiv 1(\bmod p)$$

即

$$T_0 \equiv 1(\bmod p)$$

由于 $T_1=b^{\frac{(p-1)}{2}} \bmod p$，所以

$$T_1^2 \equiv T_0 \equiv 1(\bmod p)$$

即 $T_1 \equiv 1(\bmod p)$或者 $T_1 \equiv -1(\bmod p)$

如果 $T_1 \equiv 1(\bmod p)$，由于

$$T_2^2 \equiv T_1 \equiv 1(\bmod p)$$

所以，

$$T_2 \equiv 1(\bmod p) \quad 或者 \quad T_2 \equiv -1(\bmod p)$$

一般地，如果 $T_i \equiv 1(\bmod p)$，$1 \leqslant i < s-1$，则由于

$$T_{i+1}^2 \equiv T_i \equiv 1(\bmod p)$$

所以，

$$T_{i+1} \equiv 1(\bmod p) \quad 或者 \quad T_{i+1} \equiv -1(\bmod p)$$

如果 $T_0 \equiv T_1 \equiv T_2 \equiv \cdots \equiv T_{s-1} \equiv 1(\bmod p)$

则由于 $T_s^2 \equiv T_{s-1} \equiv 1(\bmod p)$

所以，

$$T_s \equiv 1(\bmod p) \quad 或者 \quad T_s \equiv -1(\bmod p)$$

因此，对任意整数 $b>0$，如果 $\gcd(c,p)=1$，则 p 一定可以通过以 b 为基的 Rabin 测试。

定理 5-2 如果 $n>2$ 是一个奇合数，则至多有$\frac{n-1}{4}$个 b，$0<b<n$，使得 n 通过以 b 为基的 Miller-Rabin 测试。

Miller-Rabin 素数判定算法描述如下：

奇数 $n \geqslant 3$，$b_k b_{k-1} \cdots b_0$ 为$(n-1)$的二进制数，随机选取整数 $a(1<a<n)$。

```
d = 1
for (i = k;k≥0;i -- )
  {
    do
      {
        x = d;
        d = (d × d) mod n;
        if (d = 1 && x! = 1 && x! = (n - 1))
          return T
        if (bi = 1)
          d = (d × a) mod n
      }
      if (d! = 1)   return T
      return F
    }
```

如果返回 T，证明一定不是素数；如果返回 F，证明可能是素数。

随机选取 a，计算 s 次。如果 $s \geqslant 5$，每次都返回 F，则是素数的概率 > 99.99%。

在 10^{10} 整数范围内，共有 455 052 511 个素数。

5.2.5 梅森素数

马林·梅森(Marin Mersenne，1588—1648)是 17 世纪法国著名的数学家，他与大科学家伽利略、笛卡儿、费马、帕斯卡、罗伯瓦、迈多治等是密友。

1640 年 6 月，费马在给梅森的一封信中写道："在艰深的数论研究中，我发现了三个非常重要的性质。我相信它们将成为今后解决素数问题的基础"。这封信讨论了形如 2^p-1 的数(其中 p 为素数)。梅森在欧几里得、费马等人有关研究的基础上做了大量的计算、验证工作，并于 1644 年在他的《物理数学随感》一书中断言：对于 $p=2,3,5,7,13,17,19,31,67,127,257$，$2^p-1$ 是素数；而对于其他所有小于 257 的数，2^p-1 是合数。前面的 7 个数(即 2，3，5，7，13，17 和 19)属于被证实的部分，是他整理前人的工作得到的；而后面的 4 个数(即 31，67，127 和 257)属于被猜测的部分。不过，人们对其断言仍深信不疑，连大数学家莱布尼兹和哥德巴赫都认为它是对的。

虽然梅森的断言中包含若干错误，但他的工作极大地激发了人们研究素数的热情。由于梅森学识渊博，才华横溢，为人热情以及最早系统而深入地研究 2^p-1 型的数，为了纪念他，数学界就把这种数称为"梅森数"，如果梅森数为素数，则称之为"梅森素数"

梅森素数在当代具有十分丰富的理论意义和实用价值。它是发现已知最大素数最有效的途径；它推动了有"数学皇后"之称的数论研究，也促进了计算数学、程序设计技术、网格计算技术以及密码技术的发展；另外探究梅森素数的方法还可用来测试计算机硬件运算是否正确。因此，科学家们认为，对于梅森素数的探究能力如何，已在某种意义上标志着一个国家的科技水平。

挪威计算机专家奥德·斯特林德莫通过参加一个名为"因特网梅森素数大搜索"(GIMPS)的国际合作项目，最近发现了第 47 个梅森素数，该素数为"2 的 42 643 801 次方减 1"。它有 12 837 064 位数，如果用普通字号将这个巨数连续写下来，它的长度超过 50 千米！

5.2.6 RSA 的安全性

RSA 算法利用了数学中存在的一种单向性。一般说来，许多数学中的函数都有单向性。这就是说，有许多运算本身并不难，但如果你想把它倒回去，做逆运算，那就难了。RSA 算法在理论上的重大缺陷就是并不能证明分解因数绝对是如此之困难，也许日后可以找到一种能够快速分解大数的因数的算法，从而使 RSA 算法失效。

目前 RSA 算法被广泛应用于各种安全或认证领域，如 Web 服务器和浏览器信息安全 E-mail 的安全和认证、对远程登录的安全保证以及各种电子信用卡系统的核心。

与单钥加密方法比较，RSA 算法的缺点就是运算较慢。用 RSA 算法加密、解密、签名和认证都是由一系列求模幂运算组成的。在实际应用中，经常选择一个较小的公钥或者一个组织使用同一个公钥，而组织中不同的人使用不同的 n，这些措施使得加密快于解密而认证快于签名。一些快速的算法如基于快速傅里叶变换的方法可以有效减少计算步骤，但是

在实际中这些算法由于太复杂而不能广泛地使用，而且对于一些典型的密钥长度它们可能会更慢。

目前对于 RSA 算法的攻击主要有以下方式：选择密文攻击（破译密文和骗取签名）、公共模数攻击、低加密指数攻击、低解密指数攻击、定时攻击。

1. 选择密文攻击

RSA 的模指数运算能够保持输入的乘法结构，即 $E_k(ab)=E_k(a)E_k(b)(\bmod n)$，这一特点使得 RSA 能够用于消息的隐藏，适于盲签名，但它也影响了 RSA 算法的安全性。

（1）破译密文：假设 A 的公钥为 (e,n)，攻击者 B 通过窃听得到发给 A 的密文 $c=m^e \bmod n$，为了获得明文，攻击者随机选取 $r\in Z_n^*$，计算 $y=r^e \bmod n$，$t=y^c \bmod n$，令 $k=r^{-1} \bmod n$，则 $k=y^{-d} \bmod n$，现在攻击者 B 让 A 用他的私钥对 t 签名，攻击者就可以得到 $s=t^d \bmod n$，这样攻击者可以计算，$k^s\equiv y^{-d}t^d\equiv y^{-d}y^dc^d\equiv c^d\equiv m^{ed}\equiv m(\bmod n)$，于是攻击者获得了明文 m。

（2）骗取签名：攻击者 B 想要获得 A 对一则非法消息 m' 的签名，他有多种方法。

方法 1：B 首先随即选取 $x\in Z_n^*$，计算 $y=x^e \bmod n$，然后攻击者计算 $m=ym' \bmod n$，并将 m 发送给 A 以获得 A 对无害消息 m 的有效签名，A 向 B 发送签名，现在攻击者 B 计算 $m^dx^{-1} \bmod n=y^dm'^dx^{-1} \bmod n=x^{ed}m'^dx^{-1} \bmod n=m'^d \bmod n$，这样，攻击者 B 得到了 A 对非法消息 m' 的签名。

方法 2：攻击者 B 首先产生两份消息 m_1，m_2 满足 $m'=m_1m_2 \bmod n$，如果攻击者 B 能获得 A 对 m_1，m_2 的签名，那么他可以计算 $m'^d \bmod n=m_1^dm_2^d \bmod n$，从而获得 A 对非法消息 m' 的签名。

所以，为了安全起见，不要对陌生人提交的随机性文件签名，并且最好先使用单向 Hash 函数对消息进行散列运算。ISO9796 分组格式可用来防止这种攻击。

2. 公共模数攻击

如果系统中每个人拥有相同的模数 n，即使每个人采用不同的公/私钥对，系统安全仍然会受到巨大的威胁。设消息为 m，两个加密密钥分别为 e_1，e_2，且满足 $\gcd(e_1,e_2)=1$，两个密文为 $c_1=me^1 \bmod n$，$c_2=me^2 \bmod n$，由于，攻击者很容易由扩展欧几里得算法得到 r，s，满足 $re_1+se_2=1$，那么 $c_1^rc_2^s \bmod n=m^{\mathrm{re1+mse2}} \bmod n=m$。因此，在一组用户之间共享模数是不安全的。

3. 低加密指数攻击

在 RSA 加密和数字签名验证中，如果选取了较低的 e 值可以加快速度，但这是不安全的，Hastad 证明如果采用不同的模数 n，相同的公钥 e，则对 $e(e+1)/2$ 个线性相关的消息加密，系统安全会受到威胁。如果消息比较短，或者消息不相关，就不存在这个问题。如果消息相同，那么只要有一个消息就可以攻击系统。一般来讲，选取 16 位以上的素数速度比较快，而且可以阻止该攻击。对于较短的消息，使用独立随机值填充消息，可以阻止该攻击，这也能保证 $m^e \bmod n\neq m^e$。

Machael Wiener 给出了另外一种攻击，可以成功地计算解密指数 d，前提是满足以下条

件：3 $d<n^{1/4}$，并且 $q<p<2q$。

4. 定时攻击

定时攻击主要针对 RSA，核心运算是非常耗时间的模乘，只要能精确监视 RSA 的解密过程，获得解密时间，就可以估算出私有密钥 d。模指数运算是通过一位一位来计算的，每次迭代执行一次模乘，并且如果当前位是 1，则还需要进行一次模乘。对于有些密码，后一次模乘执行速度会极慢，攻击者就可以在观测数据解密时，根据执行时间判断当前位是 1 还是 0，不过这种方法只是理论上可以考虑的，实际操作很困难。如果在加密前对数据做盲化处理，再进行加密，使得加密时间具有随机性，最后进行去盲，这样可以抵抗定时攻击，不过增加了数据处理步骤。

RSA 算法的安全性主要依赖于 RSA 参数的选择，因此需要对这个算法中的各个参数仔细选择。下面介绍 RSA 参数的选择。

p，q 选择强素数，否则不能防御某些特殊的因子分解方法。假设 p，q 不是强素数。可以假设 $p-1$ 没有大的素因子，$p-1=p_1^{a1}\cdots\cdots p_m^{am}$，其中 p_i 为素数，a_i 是自然数($1\leqslant i\leqslant m$)。可以设 $p_i(1\leqslant i\leqslant m)<A$，$A$ 为一较小的整数，此时分解 n 就比较容易。设 $a\geqslant a_i(1\leqslant i\leqslant m)$，可以构造 $B=p_1^a\cdots\cdots p^{ma}$，此时必有 $(p-1)|B$。由费马定理知道，$2^B=1 \bmod p$，又因为有 $(p-1)|B$。处理 $x^B=y \bmod n$ 如下，可以把 x^B 看成 p 的某整数倍加 1。如果 $2^B=y \bmod n$ 中，$y=1$，则把 x 换成 3……直到 $y\neq 1$。那么，$\gcd(y-1,n)=p$，因为 y 应该为 p 的某整数倍加 1。由此可以求出 p 与 q。

p 与 q 之差要比较大，否则 $n\approx(p+q)^2/4-(p-q)^2/4$。也就是说 $n^{0.5}$ 接近 $(p+q)/2$，逐个找比 $n^{0.5}$ 略大的自然数 N，到使 (N^2-n) 是一个完全平方数。可以设 $x^2=N^2-n$，则 $n=N^2-x^2=(N+x)(N-x)$，则 $p=N-x$，$q=N+x$。

$p^3/1$ 与 $q^3/1$ 的最大公因子应很小，否则，RSA 有可能在不需要因子分解时即可被攻破，$p^3/1$ 与 $q^3/1$ 都应包含大的素因子。

p，q 应该足够大，使得在计算上分解 n 是不可能的。

$\gcd((p-1),(q-1))$ 小。否则，可以采用迭代方法。对密文 $C=M^e \bmod n$ 反复进行 e 次幂的运算。c^e，c^{ee}，……，到出现 c 的 et 次幂 mod n 为 c 时为止。则 c 的 e^{t-1} 次幂 mod n 为 M，当 t 不是很大时，这种攻击是有效的。由 Eluer 定理知 $et=1 \bmod \Phi(n)$，同样由 Eluer 定理，t 的最小值有 $t=\Phi(\Phi(n))=\Phi((p-1)(q-1))$，如果 $\gcd((p-1),(q-1))$ 小，$\Phi(\Phi(n))$ 就很大，t 就会很大。

e 不可选择过小，加密速度快但可以采用低指数攻击。在 $C=M^e \bmod n$ 中，如果 e 选择过小，可能没有模 n 的运算，可以通过直接开平方得到。一般选择使 $e^i=1 \bmod \Phi(n)$ 中的 i 尽可能大的 e。

密钥 d 的选取是最为关键的，应使 $d>N^{0.25}$ 且越大越好，这是因为当 d 的长度小于 N 的长度的 0.25 倍时，攻击者可能通过连分数方法在多项式时间内求出 d，而当 $d>N^{0.25}$ 时，攻击者只能采用穷举攻击法，若 d 较小，则显然破译困难远比大因子分解的难度小，系统被直接攻破的可能性较大。另外，d 又不能太接近 e，否则 RSA 密钥系统较容易被攻破，因为攻击者最喜欢从比较小的数和 e 附近进行攻击。

用户不能使用相同的模 n，否则任一用户的 n 被分解，可通过其他用户的公钥求出其

私钥。

明文 M 的熵要尽可能大，使得在已知密文的情况下，要猜测明文的内容几乎是不可能的。

由以上所述，RSA 的全部保密性依赖于 $N=p\times q$ 分解的难度计算，是一个大因子分解问题，但是，目前并不能从理论上证明这一点。而从实践的角度说，$N=p\times q$ 的分解可使系统完全被解密。再有，即使 N 未被分解，若参数选择不当，攻击者也完全可能在可以接受的时间内解密，主要原因是明文和密文以及 N 和 e 提供了另外一些破解信息。因此，破译 RSA 不可能比大因子分解更困难。

5.3 Rabin 密码系统

Rabin 的加密法可以说是 RSA 方法的特例，于 1979 年由 Rabin 所提出，其困难度是建立在 mod n 下找出平方根。

Rabin 密码体制是对 RSA 的一种修正，它有两个特点：Rabin 密码体制不是以一一对应的单向限门函数为基础的，对同一密文，可能有两个以上对应的明文；破译该体制等价于对大整数的分解。

以下针对密钥产生与加解密程序作一简介。

1. 密钥的产生

随机选择两个大素数 p、q，满足 $p\equiv q\equiv 3 \bmod 4$，即这两个素数形式为 $4k+3$；计算 $n=p\times q$。以 n 作为公钥，p、q 作为密钥。

2. 加密

计算密文 $c=m^2 \bmod n$。其中 m 为明文，c 为密文。

3. 解密

解密就是求 c 模 n 的平方根，由中国剩余定理知，解该方程组等价于解方程组。

$$\begin{cases} x^2 \equiv c \bmod p \\ x^2 \equiv c \bmod q \end{cases}$$

由于 $p\equiv q\equiv 3 \bmod 4$，方程组的解可以很容易地求出，其中每个方程都有两个解，即

$$x \equiv m \bmod p, \quad x \equiv -m \bmod p$$
$$x \equiv m \bmod q, \quad x \equiv -m \bmod q$$

组合得到 4 个同余方程组。

$$\begin{cases} x \equiv m \bmod p \\ x \equiv m \bmod q \end{cases} \quad \begin{cases} x \equiv m \bmod p \\ x \equiv -m \bmod q \end{cases}$$
$$\begin{cases} x \equiv -m \bmod p \\ x \equiv m \bmod q \end{cases} \quad \begin{cases} x \equiv -m \bmod p \\ x \equiv -m \bmod q \end{cases}$$

由中国剩余定理可解出每一方程组的解，共有4个，即每一密文对应的明文不唯一。为了有效地确定明文，可在 m 中加入某些信息，如发送者的身份号、接收者的身份号、日期和时间等。

5.4 ElGamal 密码系统

ElGamal 为目前著名的公开密钥密码系统之一，是由 ElGamal 于1985年提出的。ElGamal 密码系统可作为加解密、数字签章等之用，其安全性是建立于离散对数问题的。

1. 密钥产生

任选一个大质数 p，使得 $p-1$ 有大质因子。

任选一个 mod p 之原根 g。

公布 p 与 g。

使用者任选一私钥 $x \in Z_p$，并计算公钥 $y = g^x \bmod p$。

2. 加密

任选一个随机数 $r \in Z_p$ 满足 $\gcd(r, p-1) = 1$，并计算

$$c_1 = g^r \bmod p$$

$$c_2 = m \times y^r \bmod p \quad (m \text{ 为明文})$$

密文为 $\{c_1, c_2\}$。

3. 解密

计算 $w = (c_1^x)^{-1} \bmod p$。

计算明文 $m = c_2 \times w \bmod p$。

破解 ElGamal 公钥密码系统最直接的办法是计算离散对数。当 $p-1$ 所有的因子都是小素数时，可以采用 Pohlig-Hellman 算法。此外可以仿照因子分解算法引入因子库，先计算因子库中素数的离散对数，然后计算期望元素 β 的离散对数，这是比较有效的指标计算方法。

ElGamal 公钥密码系统可以在任何循环群上实现。选择群的标准是：群中的运算容易实现，以保证有效性；群中离散对数问题在计算上是困难的，以保证安全性。

ElGamal 方法具有以下优点：系统不需要保存秘密参数，所有的系统参数均可公开；同一个明文在不同的时间由相同加密者加密会产生不同的密文（机率式密码系统），但 ElGamal 方法的计算复杂度比 RSA 方法要大。

目前最受关注的群是定义在有限域上椭圆曲线的点所构成的循环群。椭圆曲线公钥密码系统是替代 RSA 公钥密码系统的最佳选择。它比 RSA 和有限域上的公钥密码系统更加有效。因为要达到1024位 RSA 安全水平，椭圆曲线公钥密码系统只要163位数上的运算就足够了。虽然这里运算比素数域的模运算要复杂，但是密钥要短很多。椭圆曲线群的另一个特点是对同样的基域可以选择不同的椭圆曲线，而这些不同的椭圆曲线可采用相同的

芯片来实现域的运算。

5.5 椭圆曲线密码系统

椭圆曲线密码学(Elliptic Curve Cryptography,ECC)是基于椭圆曲线数学的一种公钥密码方法。1985 年,Neal Koblitz 和 Victor Miller 分别独立提出了椭圆曲线密码体制(ECC),其依据就是定义在椭圆曲线点群上的离散对数问题的难解性。

ECC 被广泛认为是在给定密钥长度的情况下,最强大的非对称算法,因此在对带宽要求十分紧的连接中会十分有用。

ECC 的主要优势是在某些情况下比其他方法使用更小的密钥——比如 RSA——提供相当的或更高等级的安全。ECC 的另一个优势是可以定义群之间的双线性映射,基于 Weil 对或是 Tate 对;双线性映射已经在密码学中发现了大量的应用,例如基于身份的加密。不过一个缺点是加密和解密操作的实现比其他机制花费的时间长。

国家标准与技术局和 ANSI X9 已经设定了最小密钥长度的要求,RSA 和 DSA 是 1024 位,ECC 是 160 位,相应的对称分组密码的密钥长度是 80 位。NIST 已经公布了一列推荐的椭圆曲线用来保护 5 个不同的对称密钥大小(80,112,128,192,256)。一般而言,二进制域上的 ECC 需要的非对称密钥的大小是相应的对称密钥大小的两倍。

椭圆曲线密码学的许多形式有稍微的不同,所有的都依赖于被广泛承认的解决椭圆曲线离散对数问题的困难性上,对应有限域上椭圆曲线的群。

5.5.1 相关概念

无穷远元素(无穷远点,无穷远直线)平面上任意两相异直线的位置关系有相交和平行两种。引入无穷远点,是两种不同关系的统一。$AB \perp L_1$,$L_2 /\!/ L_1$,直线 AP 由 AB 起绕 A 点依逆时针方向转动,P 为 AP 与 L_1 的交点,如图 5-3 所示。$Q=\angle BAP \to \pi/2$ 则 $AP \to L_2$,可设想 L_1 上有一点 P_∞,它为 L_2 和 L_1 的交点,称之为无穷远点。直线 L_1 上的无穷远点只能有一个(因为过 A 点只能有一条平行于 L_1 的直线 L_2,而两直线的交点只能有一个)。

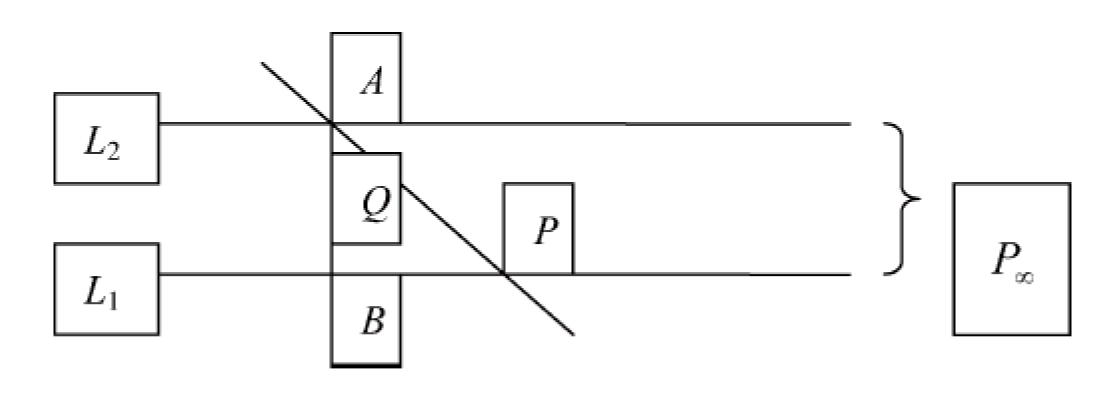


图 5-3 无穷远点

所以,平面上一组相互平行的直线,有公共的无穷远点(为与无穷远点相区别,把原来平面上的点叫做平常点)。

平面上任何相交的两直线 L_1,L_2 有不同的无穷远点。原因:若否,则 L_1 和 L_2 有公共的无穷远点 P_∞,则过两相异点 A 和 P_∞ 有相异的两直线,与公理相矛盾。

全体无穷远点构成一条无穷远直线。

欧式平面添加上无穷远点和无穷远直线,自然构成射影平面,如图 5-4 所示。

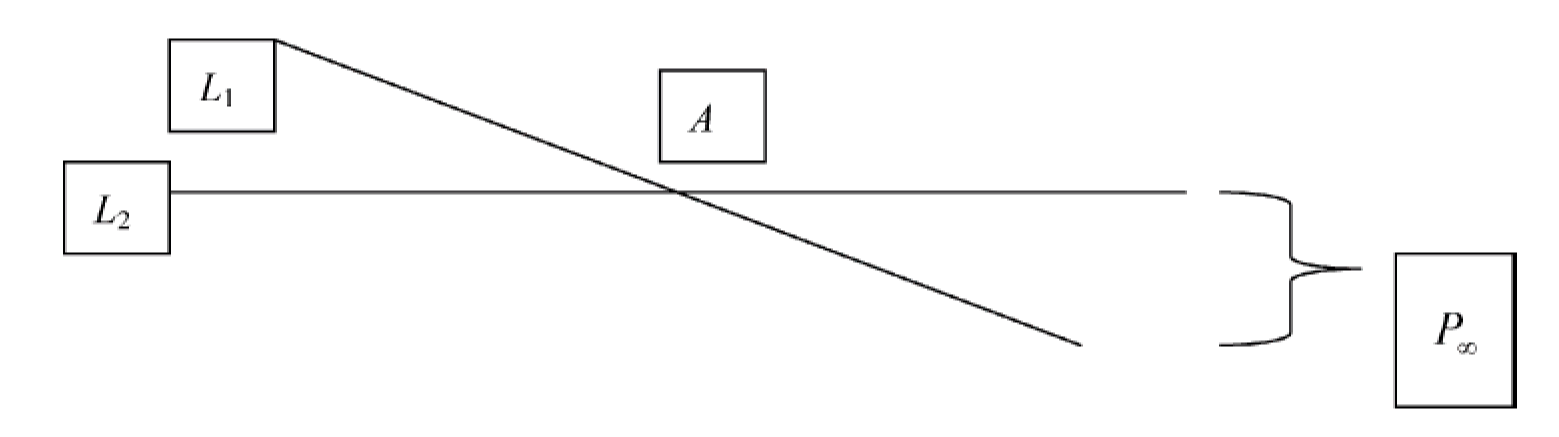


图 5-4 欧式平面

齐次坐标,解析几何中引入坐标系,用代数的方法研究欧氏空间。这样的坐标法也可推广至摄影平面上,建立平面摄影坐标系。

平面上两相异直线 L_1,L_2,其方程分别为

$$L_1: a_1x+b_1y+c_1=0$$
$$L_2: a_2x+b_2y+c_2=0$$

其中 a_1,b_1 不同时为 0;a_2,b_2 也不同时为 0。

$$设\ D=\begin{vmatrix}a_1 & b_1\\ a_2 & b_2\end{vmatrix}\quad Dx=\begin{vmatrix}b_1 & c_1\\ b_2 & c_2\end{vmatrix}\quad Dy=\begin{vmatrix}c_1 & a_1\\ c_2 & a_2\end{vmatrix}$$

若 $D\neq0$,则两直线 L_1,L_2 相交于一平常点 $P(x,y)$,其坐标为 $x=Dx/D,y=Dy/D$。这组解可表示为 $x/Dx=y/Dy=1/D$(约定:分母 Dx,Dy 有为 0 时,对应的分子也要为 0)。上述表示可抽象为(Dx,Dy,D)。

若 $D=0$,则 $L_1/\!/L_2$,此时 L_1 和 L_2 交于一个无穷远点 P_∞。这个点 P_∞ 可用过原点 O 且平行于 L_2 的一条直线 L 来指出他的方向,而这条直线 L 的方程就是:$a_2x+b_2y=0$。

为把平常点和无穷远点的坐标统一起来,把点的坐标用(X,Y,Z)表示,X,Y,Z 不能同时为 0,且对平常点(x,y)来说,有 $Z\neq0,x=X/Z,y=Y/Z$,于是有

$$\frac{X/Z}{Dx}=\frac{Y/Z}{Dy}=\frac{1}{D}$$

$X/Dx=Y/Dy=Z/D$,有更好的坐标抽象(X,Y,Z),这样对于无穷远点则有 $Z=0$ 也成立。

若实数 $p\neq0$,则(pX,pY,pZ)与(X,Y,Z)表示同一个点,实质上用$(X{:}Y{:}Z)$表示。三个分量中,只有两个是独立的,具有这种特征的坐标就叫齐次坐标。

设有欧氏直线 L,它在平面直角坐标系 Oxy 上的方程为 $ax+by+c=0$,则 L 上任一平常点(x,y)的齐次坐标为(X,Y,Z),$Z\neq0$,代入得 $ax+by+cz=0$。给 L 添加的无穷远点的坐标(X,Y,Z)应满足 $ax+by=0,z=0$;平面上无穷远直线方程自然为 $z=0$。

K 为域,K 上的摄影平面 $P^2(K)$是一些等价类的集合$\{(X{:}Y{:}Z)\}$。考虑下面的 Weierstrass 方程(次数为 3 的齐次方程):

$$y^2z+a_1xyz+a_3yz^2=x^3+a_2x^2z+a_4xz^2+a_6z^3$$

其中,系数 $a_i\in K$,或 $a_i\in K$ 为 K 的代数闭域。

Weierstrass 方程被称为光滑的或非奇异的是指对所有适合以下方程的射影点 $P=(X{:}Y{:}Z)\in P^2(K)$来说,

$$F(x,y,z)=y^2z+a_1xyz+a_3yz^2-x^3-a_2x^2z-a_4xz^2-a_6z^3=0$$

在 P 点的三个偏导数之中至少有一个不为 0,若否称这个方程为奇异的。

5.5.2 椭圆曲线

椭圆曲线 E 是一个光滑的 Weierstrass 方程在 $P^2(K)$中的全部解集合。

$$y^2z + a_1xyz + a_3yz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3$$

在椭圆曲线 E 上恰有一个点，称之为无穷远点，即(0:1:0)用 θ 表示。椭圆曲线的研究来源于椭圆积分：

$$\int \frac{\mathrm{d}x}{\sqrt{E(x)}}。$$

这里，$E(x)$是 x 的三次多项式或四次多项式。这样的积分不能用初等函数来表达，为此引进所谓的椭圆函数。所谓椭圆曲线指的是韦尔斯特拉斯(Weierstrass)方程，可用非齐次坐标的形式来表示，设 $x=X/Z$，$y=Y/Z$，于是原方程可转化为

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \tag{5-1}$$

此时，椭圆曲线 E 就是方程式(5-1)在射影平面 $P^2(K)$上的全部平常点解，外加一个无穷远点 θ 组成的集合。

若 $a_1,a_2,a_3,a_4,a_6 \in K$，此时椭圆曲线 E 被称为定义在 K 上，用 E/K 表示。如果 E 能被限定在 K 上，那么 E 的 K——有理点集合表示为 $E(K)$，它为 E 中全体有理坐标点的集合外加无穷远点 θ。

实域 $\mathbf{R}$ 上的椭圆曲线：设 $K=\mathbf{R}$，此时的椭圆曲线可表为平面中通常曲线上的点，外加无穷远点 θ。实域 $\mathbf{R}$ 上椭圆曲线的点的加法运算法则：

设 $L \in P^2(\mathbf{R})$为一条直线。因为 E 的方程是三次的，所以 L 可与 E 在 $P^2(\mathbf{R})$恰有三个交点，记为 P、Q、R(注意，如果 L 与 E 相切，那么 P、Q、R 可以不是相异的)。按下述方式定义 E 上的运算$\oplus$。

设 P、$Q \in E$，L 为连接 P、Q 的直线(若 $P=Q$，则 L 取过 P 点的切线)；设 R 为 L 与 E 的另一个交点；再取连接 R 与无穷远点的直线 L'，则 L'与 E 的另一个交点定义为 $P\oplus Q$。

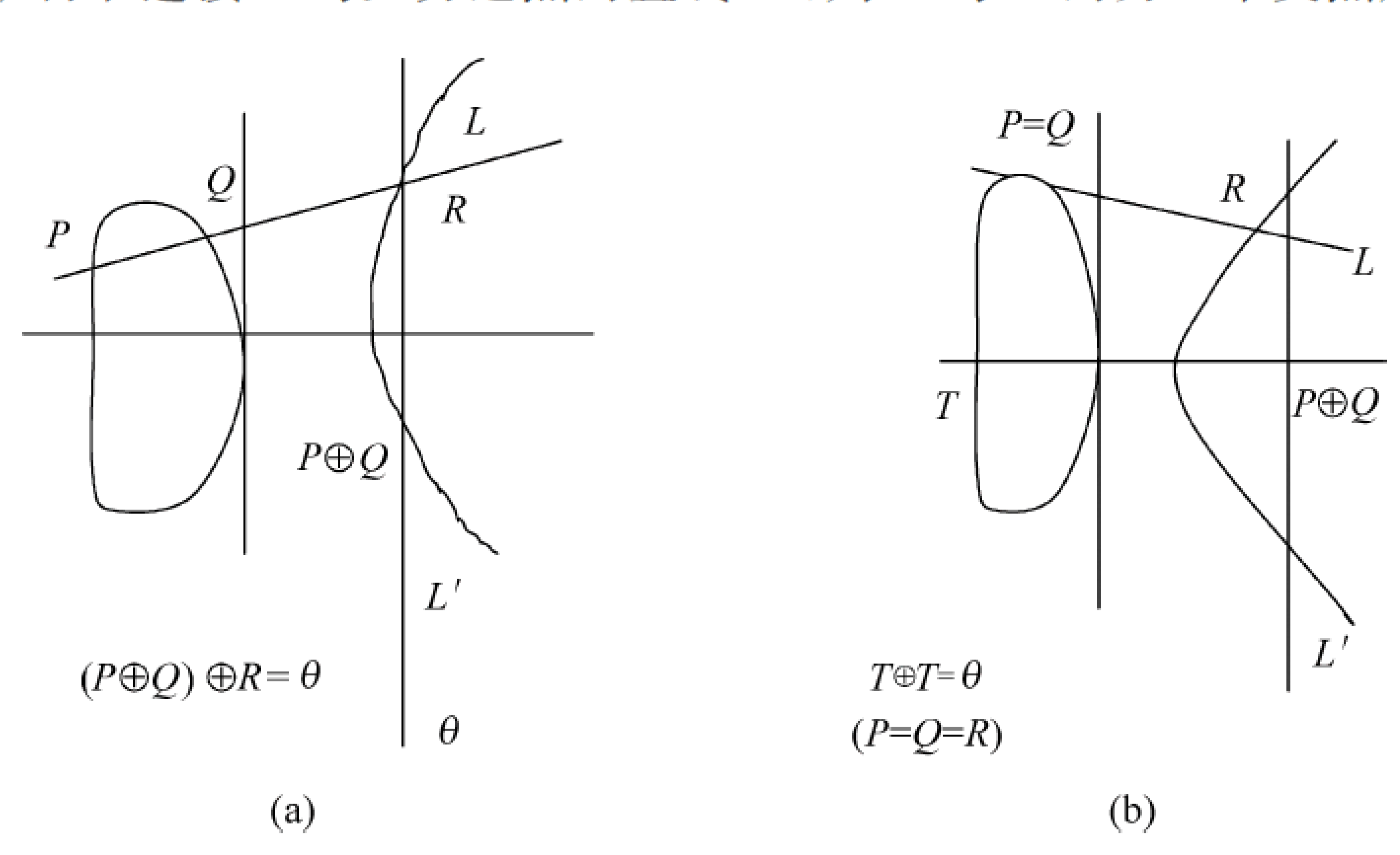


图 5-5 交点示意图

以上的实际图像为椭圆曲线 $y^2=x^3-x$ 的一般化，来自对具体曲线的抽象，对运算更具体一些。

设 $P=(x_1,y_1)$，$Q=(x_2,y_2)$，$P\oplus Q=(x_3,y_3)$，由 $P\oplus Q$ 的定义，设 $y=\alpha x+\beta$ 为通过 P，Q 两点直线 L 的方程，可算出

$$\alpha=(y_2-y_1)/(x_2-x_1),\quad \beta=y_1-\alpha x_1$$

易见，直线 L 上的一个点$(x,\alpha x+\beta)$是在椭圆曲线 E 上，当且仅当

$$(\alpha x+\beta)^2=x^3-x,\quad P\oplus Q=(x_1,y_1)\oplus(x_2,y_2)=(x_3,y_3)=(x_3,-(\alpha x_3+\beta))$$

其中，$x_3=\alpha_2-x_1-x_2=((y_2-y_1)/(x_2-x_1))2-x_1-x_2$；

$y_3=-y_1+((y_2-y_1)/(x_2-x_1))(x_1-x_3)$

当 $P=Q$ 时，$P\oplus Q=(x_3,y_3)$算得

$$x_3=((3x_12-1)/2y_1)2-2x_1;\quad y_3=-y_1+((3x_12-1)/2y_1)(x_1-x_3)$$

有结论为：

(1) 如果直线 L 与 E 相交与三点 P，Q，R(不一定相异)，那么$(P\oplus Q)\oplus R=\theta$(从图 5-5(a)中可见)；

(2) 任给 $P\in E$，$P\oplus\theta=P$(此时设 $Q=\theta$，易见 $L=L'$)；

(3) 任给 $P,Q\in E$ 有：$P\oplus Q=Q\oplus P$；

(4) 设 $P\in E$，那么可以找到$-P\in E$ 使 $P\oplus -P=\theta$；

(5) 任给 $P,Q,R\in E$，有$(P\oplus Q)\oplus R=P\oplus(Q\oplus R)$；

若 $P=Q=R$，则如图 5-5(b)所示。

综上所述，知 E 对$\oplus$运算形成一个 Abel 群。

上述规则可开拓到任意域上，特别是有限域上。假定椭圆曲线是定义在有限域 Fq 上$(q=pm)$，那么：

$E(Fq)=\{(x,y)\in Fq\times Fq \mid y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6\}\cup\{\theta\}$，它对$\oplus$形成一个群，为 Abel 群。

令 Fq 表示 q 个元素的有限域，用 $E(Fq)$表示定义在 Fq 上的一个椭圆曲线 E。$E(Fq)$的点数用$\#E(Fq)$表示，则$|\#E(Fq)-q-1|\leqslant 2q^{1/2}$。

Fp(素域，p 为素数)上椭圆曲线，令 $p>3$，$a,b\in Fp$，满足 $4a^3+27b^2\neq 0$，由参数 a 和 b 定义的 Fp 上的一个椭圆曲线方程为

$$y^2=x^3+ax+b \tag{5-2}$$

它的所有解$(x,y)(x\in Fp,y\in Fp)$，连同一个称为"无穷远点"(记为 θ)的元素组成的集合记为 $E(Fp)$，由 Hass 定理知：$p+1-2p^{1/2}\leqslant\#E(Fp)\leqslant p+1+2p^{1/2}$集合 $E(Fp)$对应下面的加法规则，且对加法$\oplus$形成一个 Abel 群：

(1) $\theta\oplus\theta=\theta$(单位元素)；

(2)$(x,y)\oplus\theta=(x,y)$，任给$(x,y)\in E(Fp)$；

(3)$(x,y)\oplus(x,-y)=\theta$，任给$(x,y)\in E(Fp)$，即点(x,y)的逆元为$(x,-y)$；

(4) 令(x_1,y_1)，(x_2,y_2)为 $E(Fp)$中的非互逆元，且满足 $x_1\neq x_2$ 的两点，则$(x_1,y_1)\oplus(x_2,y_2)=(x_3,y_3)$，其中

$$\begin{cases}x_3=\lambda^2-x_1-x_2\\ y_3=\lambda(x_1-x_3)-y_1\end{cases},\quad \lambda=\frac{y_2-y_1}{x_2-x_1} \tag{5-3}$$

(5) (倍点运算规则)设$(x_1,y_1)\in E(Fp)$，$y_1\neq 0$，则 $2(x_1,y_1)=(x_3,y_3)$，其中

$$\begin{cases} x_3 = \lambda^2 - 2x_1 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}, \quad \lambda = \frac{3x_1^2 + a}{2y_1} \tag{5-4}$$

若$\# E(Fp) = p+1$，曲线 $E(Fp)$称为超奇异的，否则称为非超奇异的。

例如：F_{23}上的一个椭圆曲线，令 $y^2 = x^3 + x + 1$ 是 F_{23}上的一个方程($a=b=1$)，则该椭圆曲线方程在 F_{23}上的解为($y^2 = x^3 + x + 1$ 的点)

(0,1),(0,22),(1,7),(1,16),(3,10),(3,13),(4,0),(5,4),(5,19),(6,4),(6,19),(7,11),(7,12),(9,7),(9,16),(11,3),(11,20),(12,4),(12,19),(13,7),(13,16),(17,3),(17,20),(18,3),(18,20),(19,5),(19,18)；θ。

群 $E(F_{23})$有 28 个点(包括无穷远点 θ)。

5.5.3 利用 ElGamal 的椭圆曲线加密法

1. 密钥产生

令系统公开参数为一个椭圆曲线 E 及模数 p。使用者执行：

任选一个整数 k，$0<k<p$。

任选一个点 $A \in E$，并计算 $B = kA$。

公钥为(A,B)，私钥为 k。

注：从(A,B)中去推导 k 相当于计算离散对数问题。

2. 加密程序

令明文 M 为 E 上的一点。首先任选一个整数 $r \in Z_p$，然后计算密文$(C_1, C_2) = (rA, M + rB)$。注：密文为两个点。

3. 解密程序

计算明文 $M = C_2 - kC_1$。

例如：$p=11$，E：$y^2 = x^3 + x + 6 \bmod 11$

$$A = (2,7), \quad k = 7 \rightarrow B = 7A = (7,2)$$

令 $M=(10,9)$，任选的 $r=3$，则

$$(C_1, C_2) = (rA, M + rB) = (3(2,7), (10,9) + 3(7,2)) = ((8,3), (10,2)) (\bmod 11)$$

解密时，计算明文如下：

$$\begin{aligned} M &= C_2 - 7C_1 = (10,2) - 7(8,3) = (10,2) - (3,5) \\ &= (10,2) + (3,6) = (10,9) (\bmod 11) \end{aligned}$$

5.5.4 利用 Menezes-Vanstone 的椭圆曲线加密法

1. 密钥产生

令系统公开参数为一个椭圆曲线 E 及模数 p。使用者执行：

任选一个整数 k，$0<k<p$。

任选一个点 $A \in E$，并计算 $B = kA$。

公钥为(A,B)，私钥为k。

2. 加密程序

令明文$M=(m_1,m_2)$不需要在E上。

任选一个数$r\in Z_H$，其中为E所包含的一个循环子群。

计算密文(C_1,C_2)，其中，

$$C_1 = rA$$

$$Y = (y_1, y_2) = rB$$

$$C_2 = (c_{21}, c_{22}) = (y_1 \times m_1 \bmod p, y_2 \times m_2 \bmod p)$$

3. 解密程序

计算$Z=(z_1,z_2)=kC_1$。

计算明文$M=(c_{21}\times z_1^{-1} \bmod p, c_{22}\times z_2^{-1} \bmod p)$。

5.5.5 椭圆曲线共享秘密推导机制

椭圆曲线共享秘密推导机制，主要是针对 IEEE 1363 中的 ECSVDP-DH、ECSVDP-DHC、ECSVDP-MQV，与 ECSVDP-MQVC，说明其中的运作机制。IEEE 1363 中定义一般密钥协议操作如下：

(1) 建立一个或多组有效领域参数，其中双方密钥对应与这些参数相关。

(2) 选择一个或更多个与领域参数有关联的有效私钥。

(3) 取得一个或更多个对方辅助的公钥。

(4) 视步骤(5)的密码运算(Cryptographic Operation)而定，决定一个适合的方法来验证公钥参数。若验证失败，则输出“无效”并且停止执行。

(5) 使用特定的密码运算于私钥及公钥，以产生一个共享秘密值。

(6) 为了要统一一把共享密钥，则要建立或统一密钥推衍参数并从共享秘密值推衍出一把共享密钥，及使用密钥推衍函数(Key Derivation Function)推衍出密钥推衍参数。

需要特别注意的是：

(1) 一密钥对的重复使用视实际而定，任何一方可能会使用一个给定的私钥/公钥对进行多次的密钥协议运算。

(2) 拥有者的鉴别：获得对方公钥可能包括公钥拥有者的鉴别，这可透过凭证或其他方法来验证。

(3) 领域参数与密钥验证：因为一个密钥协议的方法假设领域参数与公钥皆为有效，所以在该方法里并没有定义验证领域参数的方法。除非领域参数与公钥为无效情形的机率很低，否则建议执行密钥协议的步骤(4)。

(4) 密钥推衍参数：视密钥推衍函数而定，可使用的密钥推衍参数可能会有安全相关的限制，例如这些参数可包含密钥规格信息，协议相关公开信息，补充及私密信息。为了安全考虑，这些参数的说明必须明确。

5.5.6 椭圆曲线密码体制的优点

与基于有限域上离散对数问题的公钥体制(如 Diffie-Hellman 密钥交换和 ElGamal 密码体制)相比,椭圆曲线密码体制有以下优点。

1. 安全性高

攻击有限域上的离散对数问题可以用指数积分法,其运算复杂度为 $O(\exp \sqrt[3]{(\log p)(\log \log p)^2})$,其中 p 是模数(为素数)。而它对椭圆曲线上的离散对数问题并不有效。目前攻击椭圆曲线上的离散对数问题的方法只有适合攻击任何循环群上离散对数问题的大步小步法,其运算复杂度为 $O(\exp(\log \sqrt{P_{\max}}))$,其中,$P_{\max}$是椭圆曲线所形成的 Abel 群的阶的最大素因子。因此,椭圆曲线密码体制比基于有限域上的离散对数问题的公钥体制更安全。

2. 密钥量小

由攻击两者的算法复杂度可知,在实现相同的安全性能条件下,椭圆曲线密码体制所需的密钥量远比基于有限域上的离散对数问题的公钥体制的密钥量小。

3. 灵活性好

在有限域 GF(q)一定的情况下,其上的循环群(即 GF(q)$-\{0\}$)就定了。而 GF(q)上的椭圆曲线可以通过改变曲线参数,得到不同的曲线,形成不同的循环群。因此,椭圆曲线具有丰富的群结构和多选择性。

正是由于椭圆曲线具有丰富的群结构和多选择性,并可在保持和 RSA/DSA 体制同样安全性能的前提下大大缩短密钥长度(目前 160 比特足以保证安全性),因而在密码领域有着广阔的应用前景。

习题

1. 公钥密码体制的三种应用是什么?
2. 在公钥密码体制中,哪些参数是可以公开的?哪些参数是必须保密的?
3. 在使用 RSA 的公钥体制中,已截获发给某用户的密文 $C=10$,该用户的公钥 $e=5$,$n=35$,那么明文 M 等于多少?
4. 在 RSA 密码体制中,某给定用户的公钥 $e=31$,$n=3599$,那么该用户的私钥等于多少?

第6章 序列密码技术

序列密码(流密码)是密码学中一个重要的密码体制。它的起源可以追溯到20世纪20年代的Vernam体制,即古典密码学中的Vernam体制。将明文流和密文流都转化成二进制,然后进行异或处理,但是Vernam体制每次加密的密钥量都是相同的,这很容易被破译。1949年,Shannon(香农)证明了只有一次一密的密码体制是绝对安全的。这给序列密码技术以强有力的支持,序列密码就是模仿"一次一密"进行设计的。

序列密码属于"一次一密",但每次加解密的密钥是相同的,可以将其归结为分组密码的一种形式,相对于分组密码,序列密码的特点是:

(1) 分组密码以固定大小的分组作为每次处理的单元,而序列密码则是以每次一个元素(一个字母或者一个比特)作为基本单元;

(2) 分组密码的加密特性不随时间而变化,扩散性好,但是加解密处理速度慢,而序列密码是一个随时间变化的加密变换,扩散性不够好,但是加解密速度快;

(3) 分组密码相对复杂,而序列密码具有软件实现简单,同时还便于硬件实现。

6.1 序列密码模型

如何做到"一次一密"。

如果每次加密的密钥都是随机产生的,就可以保证"一次一密"。但是如果加密密钥是随机的,如何确保解密也是相同的呢?需要同步处理,即在保密通道中进行同步处理。

根据加密过程是否依赖于明文字符,序列密码分为同步和自同步两种。如果独立于明文字符叫做同步序列密码,如图6-1所示;否则叫做自同步序列密码,如图6-2所示。

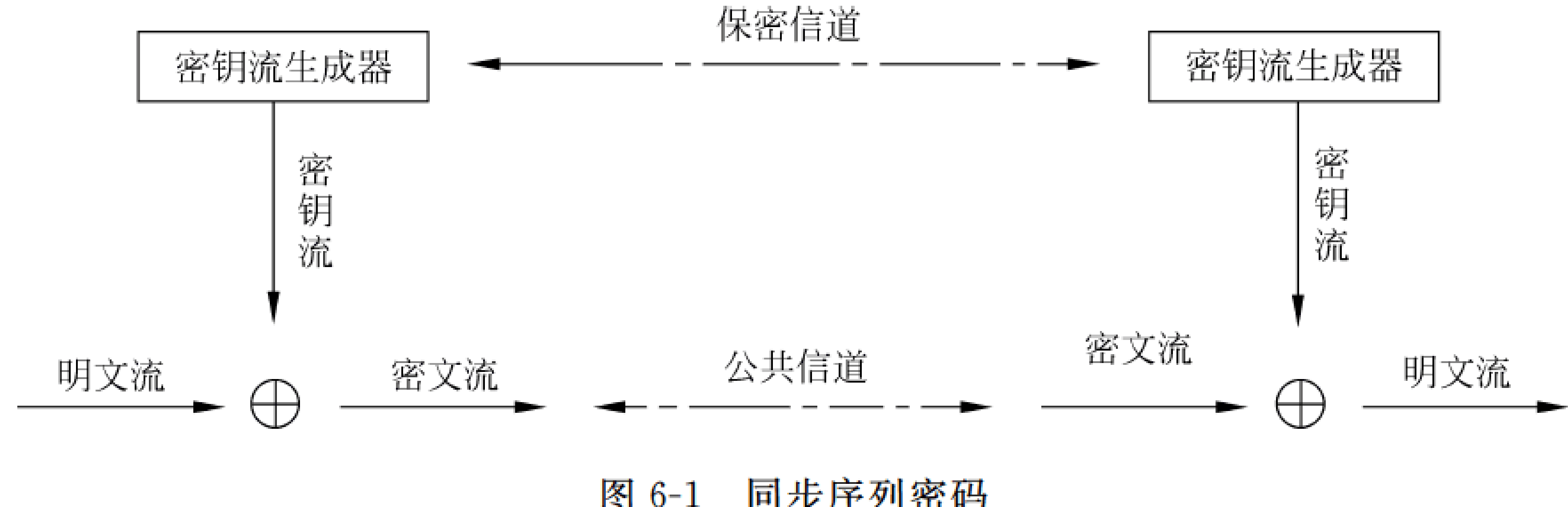


图6-1 同步序列密码

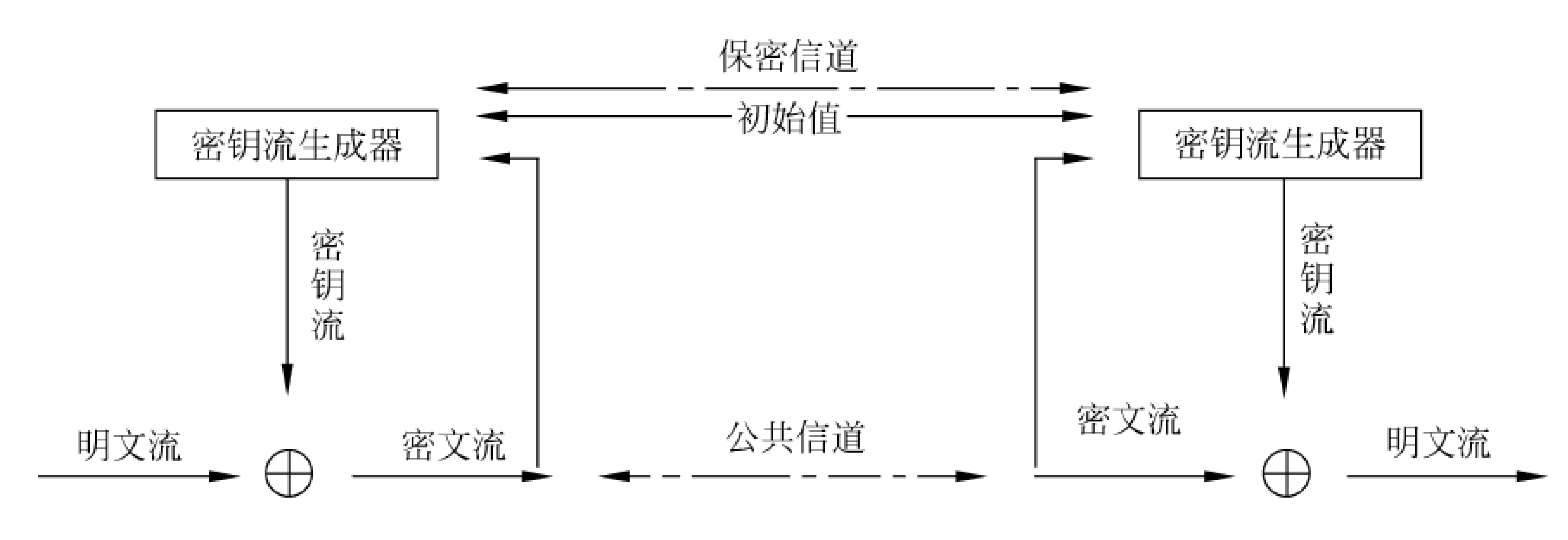


图 6-2　自同步序列密码

在同步序列密码中，如果传输过程一个密文发生丢失，那么之后的解密将失败。收发双方必须重新同步，即从丢失的密文字符开始，收发双方的密钥流生成器重新从一个相同的初态开始同步。

在自同步序列密码中，如果传输过程一个密文发生丢失，只会导致相应的若干位解密错误，不会影响后续位的解密。从这个意义上讲，自同步序列密码是优于同步序列密码的，但是自同步序列密码的密钥流与明文有关，这在实际加密中存在诸多实际问题，所以目前大多数的研究成果主要还是同步序列密码。

6.2　随机性

序列密码的安全性取决于密钥流的安全性，希望密钥流有尽可能大的周期，至少应和明文的长度相同，并且具有随机性，以使密码分析者对它无法预测。也就是说，即使截获其中一段，也无法推测后面的序列。如果密钥流是周期的，要完全做到随机是不现实的，只能要求截获比周期短的一段时不会泄露更多的信息，这样的序列称为伪随机序列。

对于 0-1 序列来说，如何定义随机性。如果掷 1000 次硬币，每次都是正面，不能称之为随机性。正反面的概率应该是差不多的，即等概率事件。

这里涉及三个概念，即周期、游程和异相自相关函数。

对序列来讲，周期是重复一个序列的最小者，如序列 001011100101110010111，周期是最小的序列 0010111，即周期为 7。

在序列的一个周期中，形如 100…001 的一段，叫做 0 游程。若 0 的个数为 r，叫做长为 r 的 0 游程，记为 0 的 r-游程。

同理，形如 011…110 的一段，叫做 1 游程。若 1 的个数为 r，叫做长为 r 的 1 游程，记为 1 的 r-游程。

在上述周期为 7 的序列 0010111 中，包含：1 个 0 的 2-游程；1 个 1 的 1-游程；1 个 0 的 1-游程；1 个 1 的 3-游程。

异相自相关函数 R_τ 可以表示为

$$R_\tau = \frac{2n_\tau}{p} - 1 \tag{6-1}$$

其中，n_τ 是延迟 τ 个比特后的序列与原序列相同比特的个数。

例如，对于序列 0010111，延迟 1 个比特后的序列为 1001011，$n_\tau = 3$。

针对伪随机序列，Golomb 提出了一个 0-1 序列(一个周期 P)的三条假设：

(1) 若 P 为偶数，则 0 和 1 的个数相同，若 P 为奇数，则 0 和 1 的个数相差 1；

(2) r-游程个数占总游程个数的 $\frac{1}{2^r}$。

(3) 异相自相关函数是一个常数。

满足这些假设并不足以说明该序列是伪随机序列，但这些是基本要求。

例 6-1 讨论一个周期内序列 010011010001100111 的随机性。

解：(1) 周期 18，0 的个数为 9，1 的个数为 9。

(2) 0 的 1 游程有 2 个，1 的 1 游程有 2 个；

0 的 2 游程有 2 个，1 的 2 游程有 2 个；

0 的 3 游程有 1 个，1 的 3 游程有 1 个；

即 1 游程共有 4 个，占总游程的比例为 $4/10 \neq 1/2$。

为了进一步证明该序列的随机性，计算异相自相关函数。

(3) $R(1)=\frac{2\times 9}{18}-1=0$

$R(2)=\frac{2\times 6}{18}-1=-\frac{1}{3}$

$R(3)=\frac{2\times 8}{18}-1=-\frac{1}{9}$

即异相自相关函数不是一个常数。所以该序列不是伪随机序列。

例 6-2 讨论一个周期内序列 1010111011000111110011010010000 的随机性。

解：(1) 周期 31，0 的个数为 15，1 的个数为 16。

(2) 0 的 1 游程有 4 个，1 的 1 游程有 4 个；

0 的 2 游程有 2 个，1 的 2 游程有 2 个；

0 的 3 游程有 1 个，1 的 3 游程有 1 个；

0 的 4 游程有 1 个，1 的 4 游程有 0 个；

0 的 5 游程有 0 个，1 的 5 游程有 1 个；

即总游程为 16，其中 1 游程共有 8 个，占总游程的比例为 1/2；

2 游程共有 4 个，占总游程的比例为 1/4；

3 游程共有 2 个，占总游程的比例为 1/8；

4 游程共有 1 个，占总游程的比例为 1/16；

5 游程共有 1 个，占总游程的比例为 1/16。

(注：按照第二条假设，5 游程应该有 0.5 个，不符合规则，所以只考虑大于 1 的情况)

(3) $R(1)=\frac{2\times 15}{31}-1=-\frac{1}{31}$

$R(2)=\frac{2\times 15}{31}-1=-\frac{1}{31}$

$R(3)=\frac{2\times 15}{31}-1=-\frac{1}{31}$

⋮

$$R(30)=\frac{2\times 15}{31}-1=-\frac{1}{31}$$

即异相自相关函数是一个常数。所以该序列是伪随机序列。

6.3 线性反馈移位寄存器

序列密码的关键是设计一个随机性好的密钥流生成器。1965 年，挪威的密码学家提出了移位寄存器理论，是随机流的主要数学工具。

移位寄存器是指具有存储数据和移位功能的寄存器，如果有 n 个存储单元，记为 n-级移位寄存器，如图 6-3 所示。所有存储单元的值可以统一向右移动一个位置，记 1 拍，图中 a_1 被首先输出。

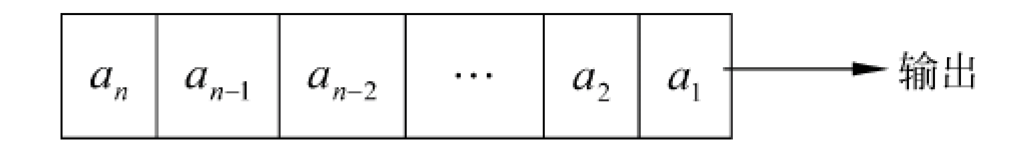


图 6-3 移位寄存器

反馈移位寄存器是由 n-级移位寄存器和一个反馈函数组成的，如图 6-4 所示。

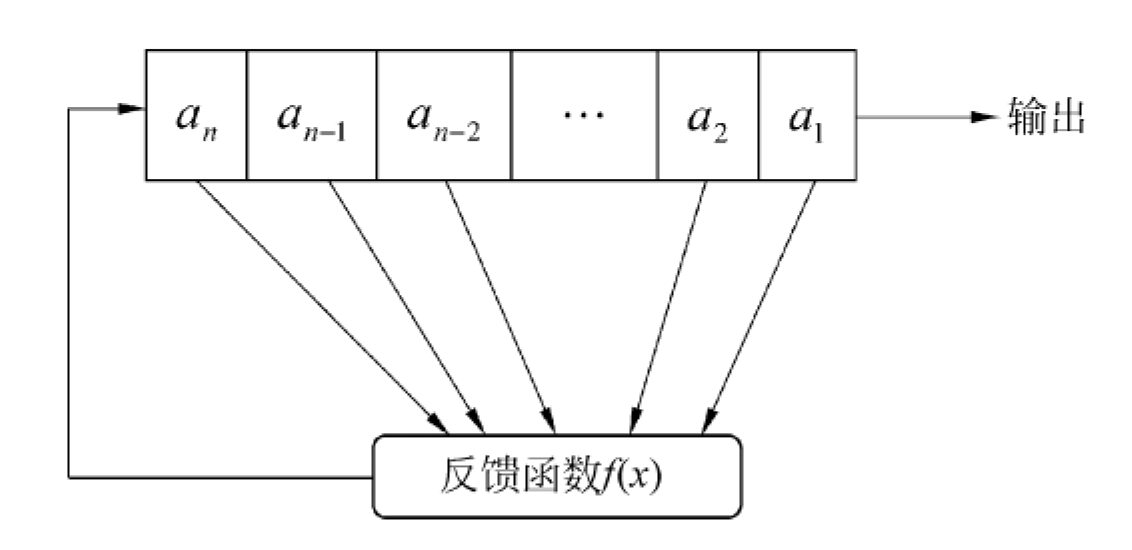


图 6-4 反馈移位寄存器

在图 6-4 中，反馈函数 $f(x)$ 由 $a_1,a_2,\cdots,a_n$ 通过某种函数关系产生 1 位，$a_1,a_2,\cdots,a_n$ 统一向右移动 1 位，a_1 被输出，由函数产生的 1 位补充到 a_n 的位置。

例 6-3 如图 6-5 所示是一个 3-级反馈移位寄存器，反馈函数 $f(x)=a_3\oplus a_2$，初态为 100，即 $a_3=1,a_2=0,a_1=0$，求其输出序列的前 8 位。

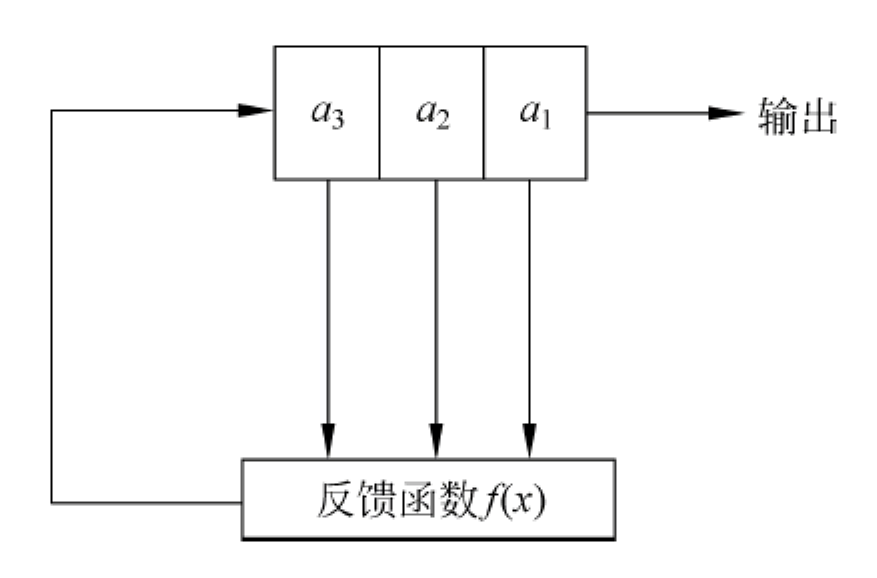


图 6-5 反馈移位寄存器

解：状态 1(初始状态)：100，反馈函数 $f(1)=a_3\oplus a_2=1$，输出：0；

状态 2：110，反馈函数 $f(2)=0$，输出：0；

状态 3：011，反馈函数 $f(3)=1$，输出：1；

状态 4：101，反馈函数 $f(4)=1$，输出：1；

状态 5：110，反馈函数 $f(5)=0$，输出：0；

状态 6：011，反馈函数 $f(6)=1$，输出：1；

状态 7：101，反馈函数 $f(7)=1$，输出：1；

状态 8：110，反馈函数 $f(8)=0$，输出：0；

所以前 8 位分别为 00110110，周期为 3，即 011。

将例 6-3 中的反馈函数推广到一般形式为

$$f(x) = c_n a_1 \oplus c_{n-1} a_2 \oplus \cdots \oplus c_1 a_n \tag{6-2}$$

其中，常数 $c_i=0,1$。c_i 可以用开关的断开和闭合来实现，如图 6-6 所示。

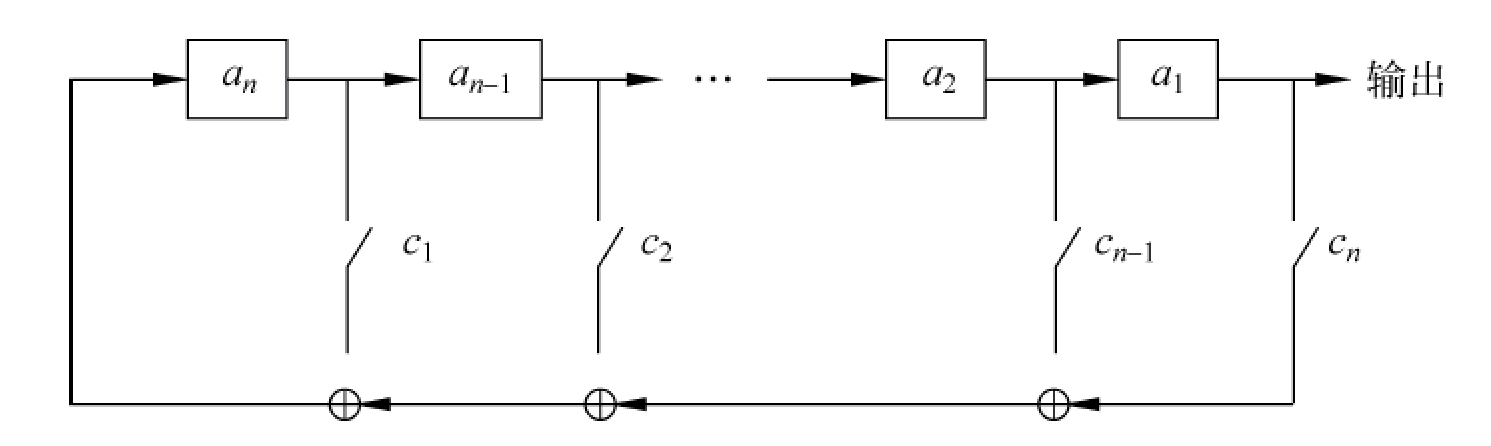


图 6-6　反馈函数的一般形式

在图 6-6 中，反馈函数是寄存器中某些位的异或，即线性形式，则把这种反馈移位寄存器称为线性反馈移位寄存器，简称 LFSR(Linear Feedback Shift Register)。而非线性反馈移位寄存器相对较复杂，不仅仅是位之间的异或，还有其他运算。

接下来讨论，对于同级的线性反馈移位寄存器，初态以及反馈函数的变化会带来什么样的变化。

例 6-4　如图 6-5 所示是一个 3-级反馈移位寄存器，反馈函数 $f(x)=a_3\oplus a_1$，初态为 100，求其输出序列的前 8 位。

解：状态 1(初始状态)：100，反馈函数 $f(1)=a_3\oplus a_1=1$，输出：0；

状态 2：110，反馈函数 $f(2)=1$，输出：0；

状态 3：111，反馈函数 $f(3)=0$，输出：1；

状态 4：011，反馈函数 $f(4)=1$，输出：1；

状态 5：101，反馈函数 $f(5)=0$，输出：1；

状态 6：010，反馈函数 $f(6)=0$，输出：0；

状态 7：001，反馈函数 $f(7)=1$，输出：1；

状态 8：100，反馈函数 $f(8)=1$，输出：0；

所以前 8 位分别为 00111010，周期为 7，即 0011101。

通过例 6-3 和例 6-4 可以看出，相同的初态，不同的反馈函数，输出序列的周期是不同的。

例 6-5　如图 6-5 所示是一个 3-级反馈移位寄存器，反馈函数 $f(x)=a_3 \oplus a_1$，初态为 011，求其输出序列的前 8 位。

解：状态 1(初始状态)：011，反馈函数 $f(1)=a_3 \oplus a_1=1$，输出：1；

状态 2：101，反馈函数 $f(2)=0$，输出：1；

状态 3：010，反馈函数 $f(3)=0$，输出：0；

状态 4：001，反馈函数 $f(4)=1$，输出：1；

状态 5：100，反馈函数 $f(5)=1$，输出：0；

状态 6：110，反馈函数 $f(6)=1$，输出：0；

状态 7：111，反馈函数 $f(7)=0$，输出：1；

状态 8：011，反馈函数 $f(8)=1$，输出：1；

所以前 8 位分别为 11010011，周期为 7，即 1101001。

可以看出，相同的反馈函数，不同的初态，输出序列的周期不因初态的改变而变化。

输出序列的周期越大，随机性越好。对于 n-级移位寄存器来说，如果给定一个初态，当输出序列遍历所有状态之后再次回到初态为最大周期。

n-级移位寄存器共有 2^n 个状态，去掉全 0 的状态，2^n-1 是最大周期。如 3-级移位寄存器有 $2^3=8$ 个状态，分别为 000，001，010，011，100，101，110，111，那么最大周期为 7。

6.4　线性移位寄存器的一元多项式表示

设 n 级线性移位寄存器的输出序列 $\{a_i\}$ 满足递推关系

$$a_{k+n} = c_1 a_{k+n-1} \oplus c_2 a_{k+n-2} \oplus \cdots \oplus c_n a_k \tag{6-3}$$

对任何 $k \geqslant 1$ 成立。这种递推关系可用一个一元 n 次多项式表示为

$$p(x) = 1 + c_1 x + c_2 x^2 + c_n x^n \tag{6-4}$$

称式(6-4)为该线性寄存器的联系多项式或特征多项式。

设 n 级线性移位寄存器对应于递推关系，则有 2^n 个递推序列，其中非恒为 0 的序列有 2^n-1 个。令这非零的序列全体为 $G[P(x)]$。对 $G[P(x)]$ 中的任一序列 a_j，有母函数 $A(x) = \sum_{i=1}^{\infty} a_i x^{i-1}$。

定理 6-1　设 $p(x)=1+c_1x+c_2x^2+\cdots+c_nx^n$ 是 GF(2)上的多项式，且递推序列 $\{a_i\} \in G[P(x)]$，令

$$A(x) = \sum_{i=1}^{\infty} a_i x^{i-1}$$

则

$$A(x) = \frac{\phi(x)}{p(x)}$$

其中

$$\phi(x) = \sum_{i=1}^{n} c_{n-i} x^{n-i} \sum_{j=1}^{j} a_j x^{j-1}$$

根据定理，若序列$\{a_t\}\in G[p_n(x)]$，其中$p_n(x)$是n级线性移位寄存器的特征多项式，则母函数为$A(x)=\phi(x)/p(x)$，其中的次数低于n，最多为$n-1$次。

定理 6-2 $p(x)|q(x)$的充要条件是$G[p(x)]\subset G[q(x)]$。

定理说明n级线性移位寄存器产生的序列可用级数更多的线性移位寄存器来实现。

定义 6-1 设$p(x)$为GF(2)上的n次多项式，使$p(x)|x^p-1$的最小p称为$p(x)$的周期或$p(x)$的阶。

定理 6-3 设$p(x)$为GF(2)上的n次多项式，且$p(x)$是序列$\{a_i\}$的特征多项式，p为$p(x)$的阶，则$\{a_i\}$的周期为$r|p$。

n级线性移位寄存器输出序列的周期r不依赖于初始条件，而依赖于特征多项式$p(x)$。

定理 6-4 若$p(x)$是n次不可约多项式，且$p(x)$的阶为m，$\{a_i\}\in G[p(x)]$，则序列$\{a_i\}$的周期为m。

定理说明了特征多项式满足什么条件，n级线性移位寄存器的输出为m序列。

定理 6-5 n级线性移位寄存器产生的状态序列最大周期为2^n-1的必要条件是其特征多项式是不可约的。

6.5 *m* 序列密码的破译

利用线性移位寄存器产生的m序列设计加密算法，其构思如图 6-7 所示。

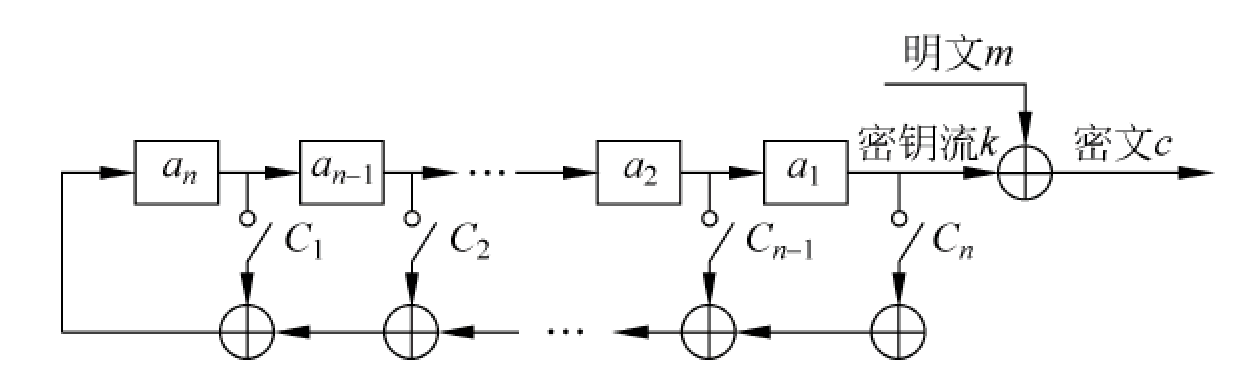


图 6-7 *m* 序列设计加密算法

算法的密钥取决于初始状态和$c_1,c_2,\cdots,c_n$的值。

n级线性移位寄存器对应的多项式是本原多项式，有$\lambda(n)$个，非 0 初始状态有2^n-1个，故不同的密钥或密钥流共有$\lambda(n)(2^n-1)$个。

特征多项式$p(x)$决定了线性移位寄存器输出序列的性质。当特征多项式为本原多项式时，线性移位寄存器输出序列周期最长为2^n-1，随机性良好。但是，线性移位寄存器在已知明文攻击时是可破译的。

设$\boldsymbol{S}_h$和$\boldsymbol{S}_{h+1}$表示线性移位寄存器输出序列任意连续的两个向量。

$$\boldsymbol{S}_h=\begin{bmatrix}a_h\\a_{h+1}\\\vdots\\a_{h+n-1}\end{bmatrix},\quad \boldsymbol{S}_{h+1}=\begin{bmatrix}a_{h+1}\\a_{h+2}\\\vdots\\a_{h+n}\end{bmatrix}$$

假定序列$\{a_j\}$满足线性递推关系

$$a_{h+n}=C_1a_{h+n-1}\oplus C_2a_{h+n-2}\oplus\cdots\oplus C_na_h \tag{6-5}$$

式(6-5)可以表示成

$$\begin{bmatrix} a_{h+1} \\ a_{h+2} \\ \vdots \\ a_{h+n-1} \\ a_{h+n} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ C_n & C_{n-1} & C_{n-2} & \cdots & C_1 \end{bmatrix} \begin{bmatrix} a_h \\ a_{h+1} \\ \vdots \\ a_{h+n-2} \\ a_{h+n-1} \end{bmatrix}$$

或

$$\boldsymbol{S}_{h+1} = \boldsymbol{M}\boldsymbol{S}_h$$

式中

$$\boldsymbol{M} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ C_n & C_{n-1} & C_{n-2} & \cdots & C_1 \end{bmatrix}$$

矩阵 $\boldsymbol{M}$ 称为反馈多项式 $p(x)=1+C_1x+C_2x^2+\cdots+C_nx^n$ 的伴侣矩阵,$\boldsymbol{M}$ 和 $p(x)$可以互相确定。

假定破译者知道了一段长 $2n$ 位的明密文对,即已知

$$m = m_1m_2\cdots m_{2n}, \quad c = c_1c_2\cdots c_{2n}$$

于是,可求出一段长 $2n$ 位的密钥序列

$$k = k_1k_2\cdots k_{2n}$$

其中 $k_i=m_i\oplus c_i$。

由此可推出线性移位寄存器的连续 $n+1$ 个状态:

$$\begin{aligned} S_1 &= (k_1 \quad k_2 \quad \cdots \quad k_n)' = (a_1 \quad a_2 \quad \cdots \quad a_n)' \\ S_2 &= (k_2 \quad k_3 \quad \cdots \quad k_{n+1})' = (a_2 \quad a_3 \quad \cdots \quad a_{n+1})' \\ &\vdots \\ S_n &= (k_{n+1} \quad k_{n+2} \quad \cdots \quad k_{2n})' = (a_{n+1} \quad a_{n+2} \quad \cdots \quad a_{2n})' \end{aligned}$$

作矩阵

$$\boldsymbol{X} = (S_1 \quad S_2 \quad \cdots \quad S_n)$$

因为

$$\begin{aligned} (a_{n+1} \quad a_{n+2} \quad \cdots \quad a_{2n}) &= (C_n \quad C_{n-1} \quad \cdots \quad C_1)\begin{bmatrix} a_1 & a_2 & \cdots & a_n \\ a_2 & a_3 & \cdots & a_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ a_n & a_{n+1} & \cdots & a_{2n} \end{bmatrix} \\ &= (C_n \quad C_{n-1} \quad \cdots \quad C_1)\boldsymbol{X} \end{aligned}$$

若 $\boldsymbol{X}$ 可逆,则

$$(C_n \quad C_{n-1} \quad \cdots \quad C_1) = (a_{n+1} \quad a_{n+2} \quad \cdots \quad a_{2n})\boldsymbol{X}^{-1}$$

例 6-6 假设破译者得到密文串 101101011110010 和相应的明文串 011001111111001。同时假定攻击者也知道密钥流是使用 5 级线性移位寄存器产生的,试破译该密码系统。

解：由明文(15 位)、密文(15 位)对可求出长为 15 位的密钥序列。

m_i	0	1	1	0	0	1	1	1	1	1	1	1	0	0	1
c_i	1	0	1	1	0	1	0	1	1	1	1	0	0	1	0
$k_i=m_i\oplus c_i$	1	1	0	1	0	0	1	0	0	0	0	1	0	1	1

由开始的 10 个密钥流比特得到上述矩阵方程。

$$(a_6\quad a_7\quad a_8\quad a_9\quad a_{10})=(C_5\quad C_4\quad C_3\quad C_2\quad C_1)\begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_2 & a_3 & a_4 & a_5 & a_6 \\ a_3 & a_4 & a_5 & a_6 & a_7 \\ a_4 & a_5 & a_6 & a_7 & a_8 \\ a_5 & a_6 & a_7 & a_8 & a_9 \end{bmatrix}$$

推出线性移位寄存器的连续 6 个状态为

$$S_1=(k_1\quad k_2\quad k_3\quad k_4\quad k_5)=(a_1\quad a_2\quad a_3\quad a_4\quad a_5)=(1\quad 1\quad 0\quad 1\quad 0)$$

$$S_2=(k_2\quad k_3\quad k_4\quad k_5\quad k_6)=(a_2\quad a_3\quad a_4\quad a_5\quad a_6)=(1\quad 0\quad 1\quad 0\quad 0)$$

$$S_3=(k_3\quad k_4\quad k_5\quad k_6\quad k_7)=(a_3\quad a_4\quad a_5\quad a_6\quad a_7)=(0\quad 1\quad 0\quad 0\quad 1)$$

$$S_4=(k_4\quad k_5\quad k_6\quad k_7\quad k_8)=(a_4\quad a_5\quad a_6\quad a_7\quad a_8)=(1\quad 0\quad 0\quad 1\quad 0)$$

$$S_5=(k_5\quad k_6\quad k_7\quad k_8\quad k_9)=(a_5\quad a_6\quad a_7\quad a_8\quad a_9)=(0\quad 0\quad 1\quad 0\quad 0)$$

$$S_6=(k_6\quad k_7\quad k_8\quad k_9\quad k_{10})=(a_6\quad a_7\quad a_8\quad a_9\quad a_{10})=(0\quad 1\quad 0\quad 0\quad 0)$$

故上述矩阵方程可写为

$$(0\quad 1\quad 0\quad 0\quad 0)=(C_5\quad C_4\quad C_3\quad C_2\quad C_1)\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

所以

$$(C_5\quad C_4\quad C_3\quad C_2\quad C_1)=(0\quad 1\quad 0\quad 0\quad 0)\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}^{-1}$$

又因

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}^{-1}=\begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

从而得

$$(C_5\quad C_4\quad C_3\quad C_2\quad C_1) = (0\quad 1\quad 0\quad 0\quad 0)\begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$
$$= (1\quad 0\quad 0\quad 1\quad 0)$$

由此可得该密码系统的密钥流产生的迭代公式为

$$a_{i+5} = C_5 a_i \oplus C_2 a_{i+3}$$

6.6 非线性反馈移位寄存器

非线性反馈移位寄存器主要包括 Geffe 序列生成器、*J-K* 触发器、Press 生成器和钟控生成器等形式。

1. Geffe 序列生成器

Geffe 序列生成器由三个 LFSR 组成，其中 LFSR2 作为控制生成器使用，如图 6-8 所示。

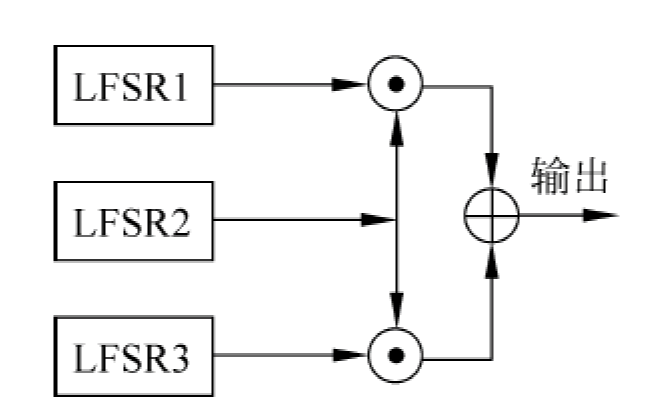


图 6-8　Geffe 序列生成器

当 LFSR2 输出 1 时，LFSR2 与 LFSR1 相连；当 LFSR2 输出 0 时，LFSR2 与 LFSR3 相连。若设 LFSRi 的输出序列为 $\{a_k^{(i)}\}(i=1,2,3)$，则输出序列 $\{b_k\}$ 可以表示为

$$b_k = a_k^{(1)} a_k^{(2)} + a_k^{(3)} \overline{a_k^{(2)}} = a_k^{(1)} a_k^{(2)} + a_k^{(3)} a_k^{(2)} + a_k^{(3)} \tag{6-6}$$

Geffe 序列生成器也可以表示为图 6-9 的形式，其中 LFSR1 和 LFSR3 作为多路复合器的输入，LFSR2 控制多路复合器的输出。

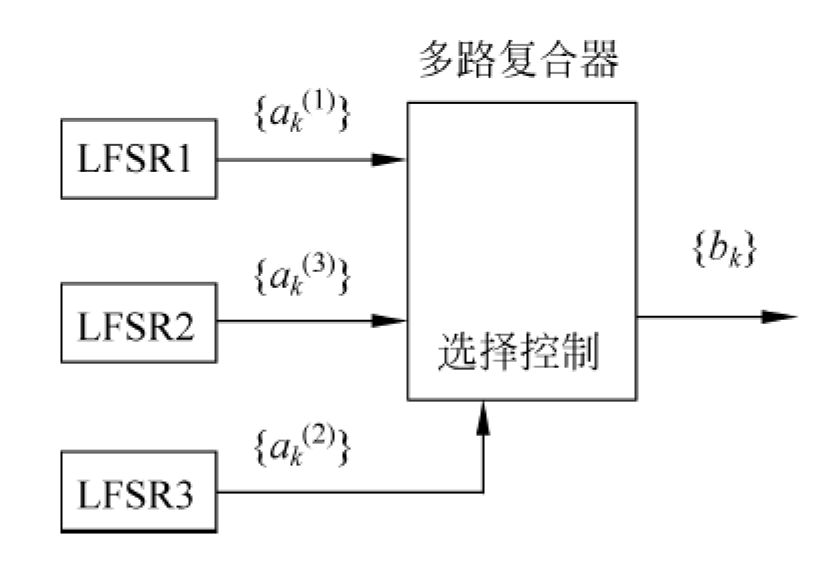


图 6-9　多路复合器表示的 Geffe 序列生成器

2. *J-K* 触发器

J-K 触发器如图 6-10 所示，它的两个输入端分别用 *J* 和 *K* 表示，其输出 c_k 可表示为

$$c_k = \overline{(x_1 + x_2)} c_{k-1} + x_1 \tag{6-7}$$

其中，x_1 和 x_2 分别是 *J* 和 *K* 端的输入。由式(6-7)可以得到 *J-K* 触发器的真值表，如表 6-1 所示。

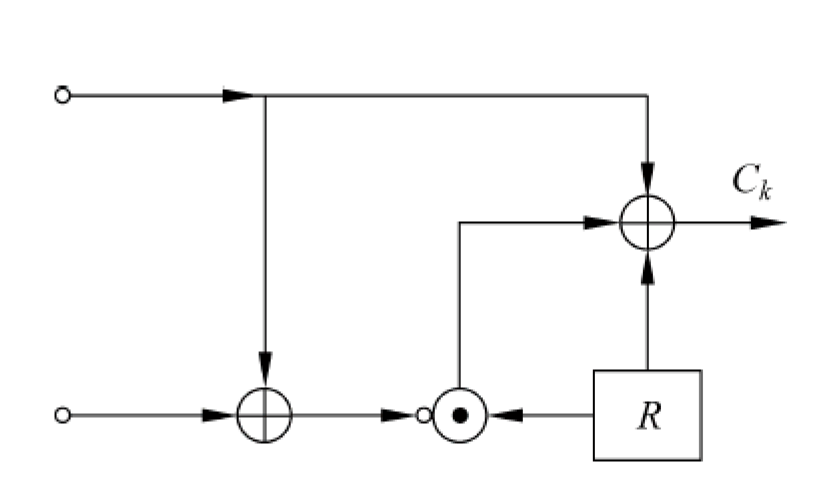


图 6-10　*J-K* 触发器

表 6-1　*J-K* 触发器真值表

J	***K***	c_k
0	0	c_{k-1}
0	1	0
1	0	1
1	1	$\overline{c_{k-1}}$

利用 *J-K* 触发器的非线性序列生成器如图 6-11 所示。

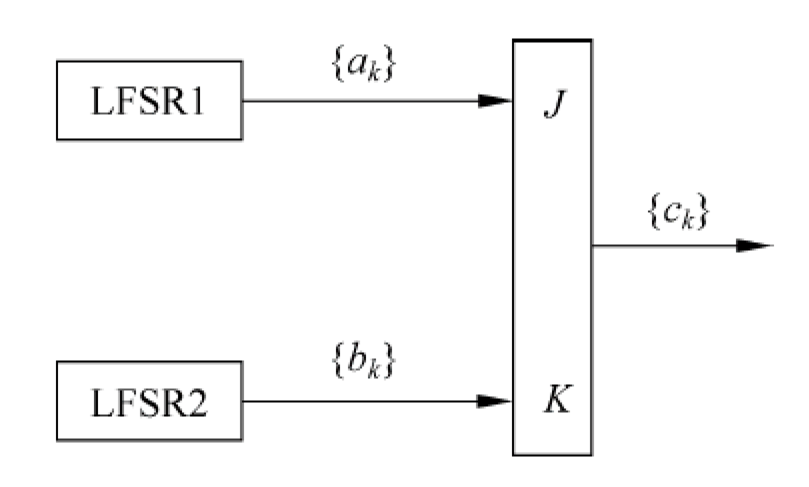


图 6-11　利用 *J-K* 触发器的非线性序列生成器

LFSR1 是 m 级线性反馈移位寄存器，LFSR2 是 n 级反馈移位寄存器。c_k 可表示为

$$c_k = \overline{(a_k + b_k)} c_{k-1} + a_k = (a_k + b_k + 1) c_{k-1} + a_k \tag{6-8}$$

由于 *J-K* 触发器的输出有时和前一项有关，通常令 $c_{-1} = 0$，则输出序列的最初三项为

$$c_0 = a_0$$
$$c_1 = (a_1 + b_1 + 1) a_0 + a_1$$
$$c_2 = (a_2 + b_2 + 1)((a_1 + b_1 + 1) a_0 + a_1) + a_2$$

当 m 与 n 互素且 $a_0 + b_0 = 1$ 时，序列$\{c_k\}$的周期为$(2^m - 1)(2^n - 1)$。

这种体制虽然在随机性方面比较好，然而只要知道它的序列的一部分，就可能求出其他部分。

为了克服这一缺点，Pless 提出了由多个 *J-K* 触发器序列驱动的多路符合序列方案，成为 Press 生成器。

3. Press 生成器

Press 生成器由 8 个线性移位寄存器组成的 4 组 *J-K* 触发器，以及一个循环计数器构

成，如图 6-12 所示。

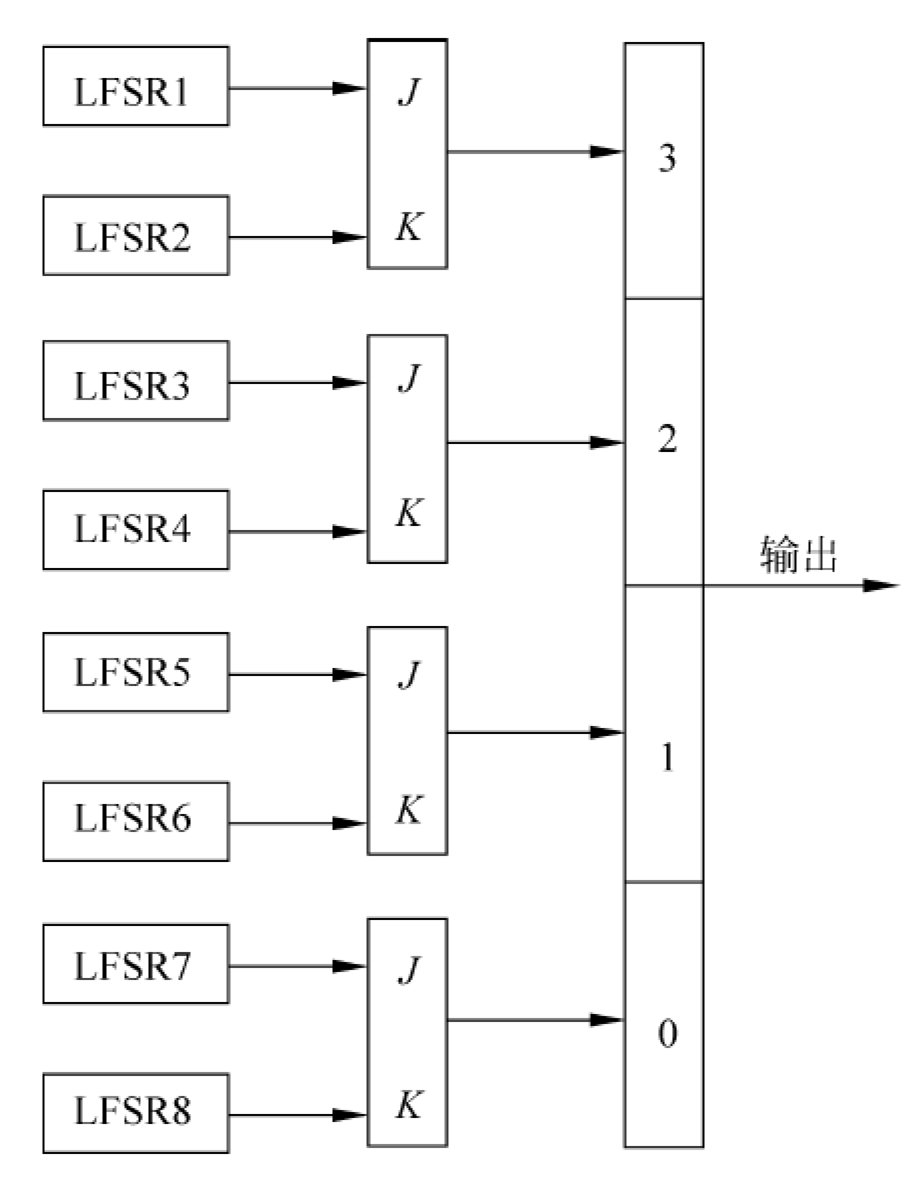


图 6-12　Press 生成器

循环计数器的作用是决定在每一个时间脉冲的作用下输出的单位。Press 生成器的密钥是 8 个移位寄存器和它们的初态、*J-K* 触发器的初态以及输出单元的顺序。

Press 生成器中 8 个线性移位寄存器的级数，不仅要求各对之间达到级数互素，而且各 *J-K* 触发器的输出周期元素要使得最后输出的周期为各个周期的乘积。

4. 钟控生成器

钟控生成器是由控制序列（由一个或多个一位寄存器来控制生成）组成的。控制序列的当前值确定被采样序列寄存器的时钟脉冲数目。控制序列和被采样序列可以是源于一个 LFSR 的自控型，也可以是源于不同 LFSR 的他控型，还可以是相互控制的互控型。

交错停走生成器是一种钟控生成器。这个生成器使用了三个不同级数的 LFSR。当 LFSR1 的输出是 1 时，LFSR2 被时钟驱动；当 LFSR1 的输出是 0 时，LFSR3 被时钟驱动。这个生成器的输出是 LFSR2 和 LFSR3 输出的异或，如图 6-13 所示。最后的输出作为密钥流的组成部分。这个生成器具有长的周期和大的线性复杂性。

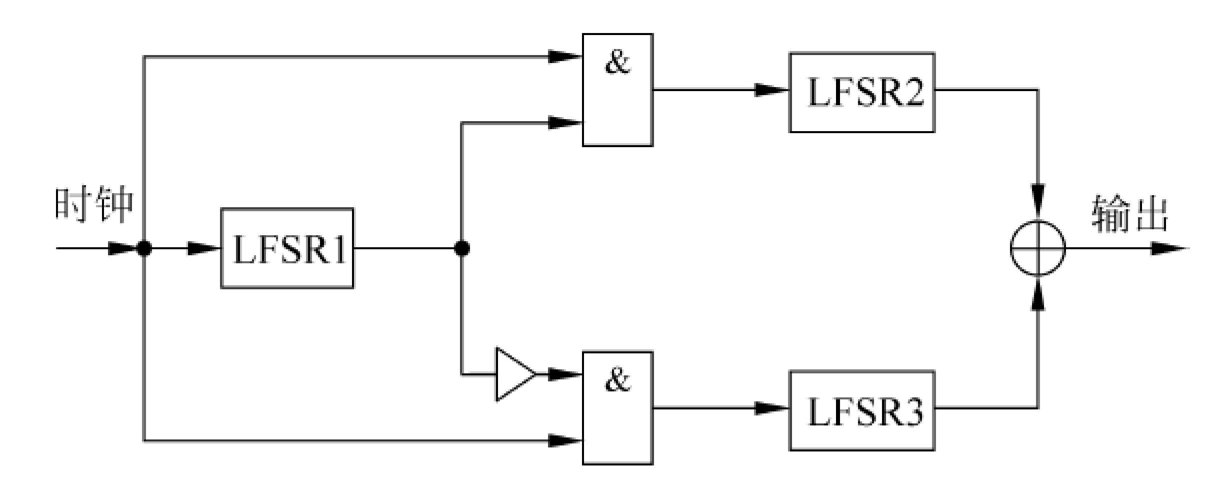


图 6-13　交错停走生成器

6.7 基于 LFSR 的序列密码加密体制

基于 LFSR 的序列密码加密体制如图 6-14 所示，明文以比特的形式进入加密系统，每一比特与反馈移位寄存器运算，得到每一比特的密文。密文同时作为移位寄存器的下一个输入。基于 LFSR 的序列密码解密体制如图 6-15 所示，移位寄存器不变，明文和密文的顺序正好相反。

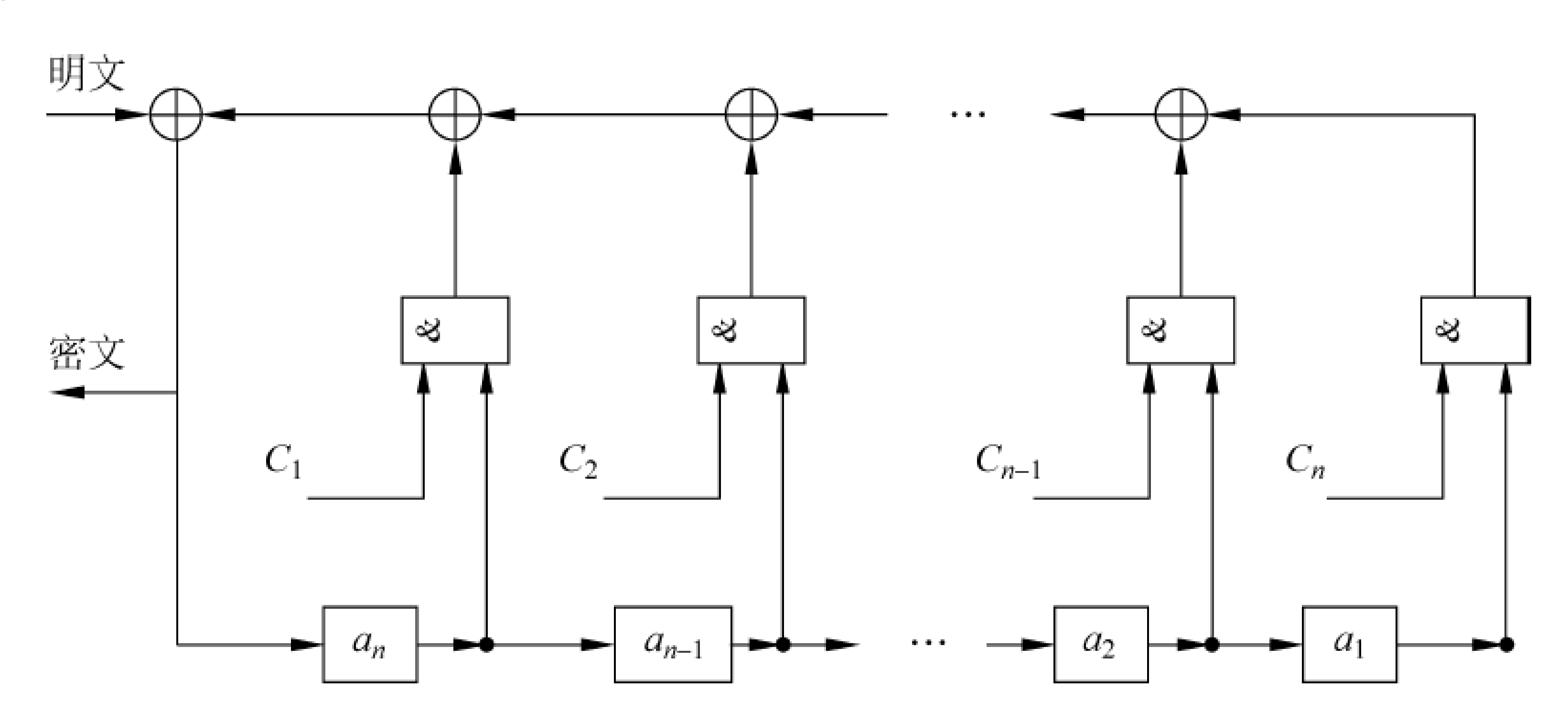


图 6-14 基于 LFSR 的序列密码加密体制

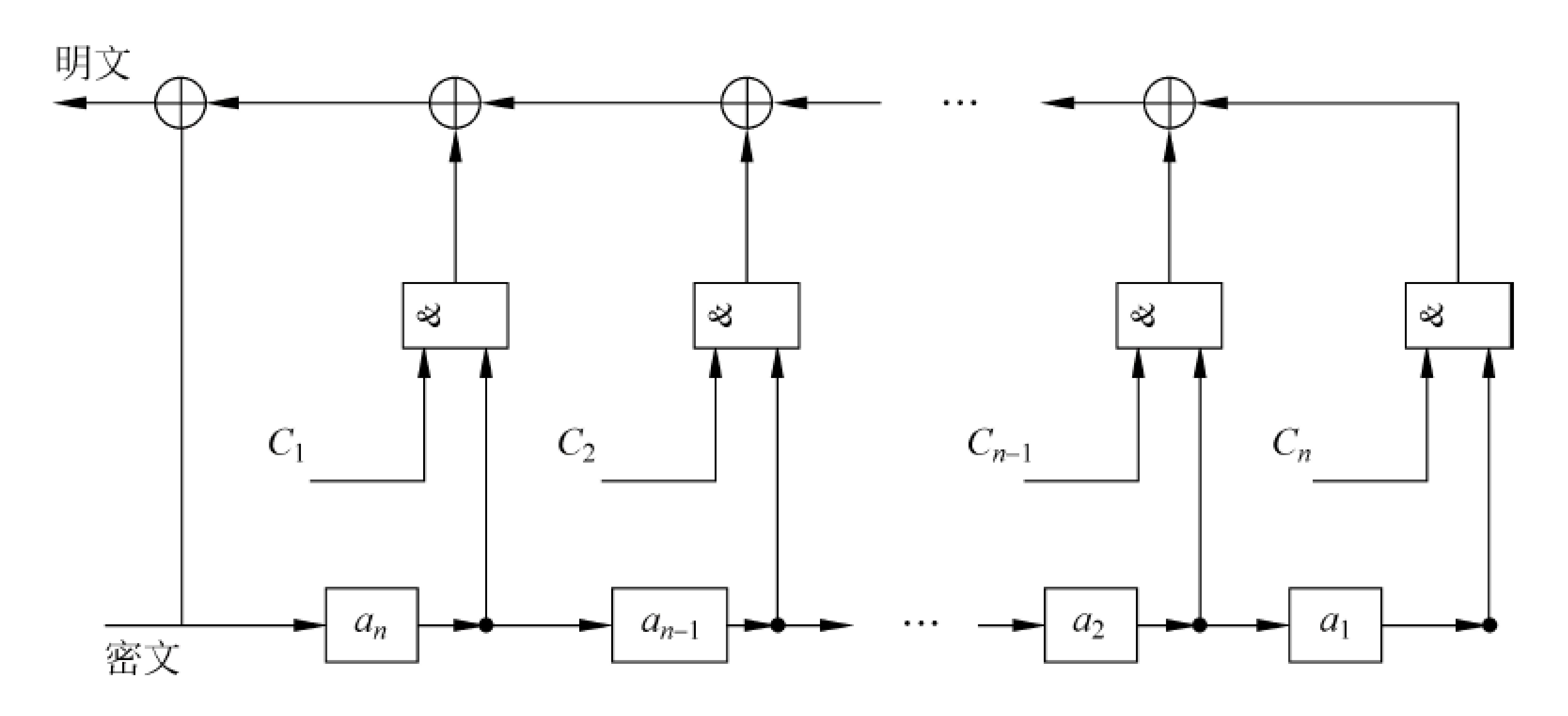


图 6-15 基于 LFSR 的序列密码解密体制

例 6-7 利用图 6-13 的加密体制加密明文(m)字母 A，10-级反馈移位寄存器，其中参数及初态分别为($c_1c_2\cdots c_{n-1}c_n$)=(1000111001)，($a_na_{n-1}\cdots a_2a_1$)=(0011101001)，求密文(e)。

解：

(1) 将字母 A 转化为 ASCII 为 65，进而转化成二进制为 01000001。

(2) 由参数及初态可知：反馈函数 $f(x)=a_{10}\oplus a_6\oplus a_5\oplus a_4\oplus a_1$。

(3) 明文 m 第一个进入系统的是 $m_1=0$；$f(1)=1$；密文 $e_1=1$。

(4) $m_2=1$；$f(2)=1$；密文 $e_2=0$。

(5) $m_3=0$；$f(3)=1$；密文 $e_3=1$。

(6) $m_4=0$；$f(4)=0$；密文 $e_4=0$。

(7) $m_5=0$；$f(5)=1$；密文 $e_5=1$。

(8) $m_6=0$；$f(6)=1$；密文 $e_6=1$。

(9) $m_7=0$；$f(7)=1$；密文 $e_7=1$。

(10) $m_8=1$；$f(8)=0$；密文 $e_8=1$。

所以密文为 10101111。

6.8　随机数产生器的安全性评估

评价随机数产生器的优劣主要从以下两个方面进行衡量：周期是否足够大；是否具有不可预测性。评估的方法主要有：统计测试，包括 Chi-Square 测试法和 Kolmogorov-Smirnov 测试法；线性复杂度测试。

1. Chi-Square 测试法

Chi-Square(卡方)测试法是一种用途很广的计数资料的假设检验方法。它属于非参数检验的范畴，主要是比较两个及两个以上样本率(构成比)以及两个分类变量的关联性分析。其根本思想就在于比较理论频数和实际频数的吻合程度或拟合优度问题。在序列密码中，主要是测试输出序列的概率分布，是否接近给定的概率分布函数。在一般应用场合中，常假设此种概率分布为均匀分布。

在掷骰子的试验中，若设 n 为测试的总次数；i 为骰子的点数；Y_i 为 i 出现的次数；P_i 为 i 出现的概率，用 Chi-Square 测试法得到的测试值如式(6-9)所示。

$$V=\sum_{1\leqslant i\leqslant k}\frac{(Y_i-np_i)^2}{np_i} \tag{6-9}$$

其中，$Y_1+Y_2+\cdots+Y_k=n$，$p_1+p_2+\cdots+p_k=1$。

为了使 Chi-Square 测试能更准确，n 值必须足够大，一般而言，n 要大到使 np_i 至少为 5 或更大。测试最好做两次以上，而且每次取不同的样本，这样判定能更准确。

2. Kolmogorov-Smirnov(柯尔莫诺夫-斯米尔诺夫)测试法

Chi-Square 测试法主要测试输出序列的概率分布，在整体上判定是否接近给定的概率分布函数，而 Kolmogorov-Smirnov 测试法主要是在区域上是否接近给定的概率分布函数。

Kolmogorov-Smirnov 测试法基于累计分布函数，用以检验两个经验分布是否不同或一个经验分布与另一个理想分布是否不同。

Kolmogorov 分布是随机变量的一种分布。

$$K=\sup_{t\in[0,1]}|B(t)| \tag{6-10}$$

其中，$B(t)$是 Brown 桥。K 的累积分布函数由下式给出：

$$P_r(K\leqslant x)=1-2\sum_{i=1}^{\infty}(-1)^{i-1}\mathrm{e}^{-2i^2x^2}=\frac{\sqrt{2\pi}}{x}\sum_{i=1}^{\infty}\mathrm{e}^{-(2i-1)^2\pi^2/(8x^2)} \tag{6-11}$$

3. 线性复杂度

线性复杂度是指对于一串序列 $s_0,s_1,\cdots,s_{N-1}$，能够用最少移位寄存器的 LFSR 产生此

序列时，移位寄存器的数目。此数目称为此序列的线性复杂度。一般而言，对于任意周期性的序列而言，若其线性复杂度 L 已知，则可以很容易地从 $2L$ 个输出序列($a_{n-L},\cdots,a_n,\cdots,a_{n+l-1}$)求出特征多项式的系数 $c_1,c_2,\cdots,c_L$。

$$\begin{bmatrix} a_{n-1} & a_{n-2} & \cdots & a_{n-L} \\ a_n & a_{n-1} & \cdots & a_{n-L+1} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n+L-2} & a_{n+L-3} & \cdots & a_{n-1} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_L \end{bmatrix} = \begin{bmatrix} a_n \\ a_{n+1} \\ \vdots \\ a_{n+L-1} \end{bmatrix}$$

所以，可以利用已知明文攻击法对此序列进行攻击。为了防止被攻击，可以将序列加密算法使用非线性组合而增加其线性复杂度。然而，对于任意的线性周期序列所做的非线性组合还是周期性的，因此可以用 LFSR 来实现。并利用 Berlekamp-Massey 算法找出其线性复杂度及其特征多项式的系数。一般而言，输出序列的线性复杂度越大越好。

总之，设计一个性能良好的序列密码是一项十分困难的任务，最基本的设计原则是“密钥生成器的不可预测性”，它可分解为众多基本原则：长周期、高线性复杂度、统计性能良好、足够“混乱”、足够“扩散”以及抵抗不同形式的攻击等。

6.9 序列密码的攻击方法

插入攻击法是序列密码的一个比较典型的攻击方法。

插入攻击法的攻击需求是可以在明文流插入一位，并截获密文流。假设原始的明文流、密钥流和密文流分别为

$$\begin{gathered} p_1,p_2,p_3,p_4,p_5,\cdots \\ k_1,k_2,k_3,k_4,k_5,\cdots \\ c_1,c_2,c_3,c_4,c_5,\cdots \end{gathered} \tag{6-12}$$

攻击者可以在明文流中插入一个已知位 p。如插入在第一位的后面，用同样的密钥加密后发送，得到的明文流、密钥流和密文流分别为

$$\begin{gathered} p_1,p_2,p_3,p_4,p_5,\cdots \\ k_1,k_2,k_3,k_4,k_5,k_6,\cdots \\ c_1,c_2,c_3,c_4,c_5,c_6,\cdots \end{gathered} \tag{6-13}$$

由于攻击者知道 p 和密文流 $c_1,c_2,c_3,c_4,c_5,c_6,\cdots$，所以可以通过建立方程组来求解。

在式(6-10)中，可以求出 k_2。

$$p \oplus k_2 = c_2 \Rightarrow k_2 = p \oplus c_2 \tag{6-14}$$

在式(6-9)中，用 k_2 可以求出 p_2。

$$p_2 \oplus k_2 = c_2 \Rightarrow p_2 = k_2 \oplus c_2 \tag{6-15}$$

在式(6-10)中，用 p_2 可以求出 k_3。

$$p_2 \oplus k_3 = c_3 \Rightarrow k_3 = p_2 \oplus c_3 \tag{6-16}$$

同理，可以依次求出 $k_4,k_5,\cdots$；$p_3,p_4,\cdots$。

例如，原始的明文密文流为

$$p_1,p_2,p_3,p_4,p_5,\cdots = 10110\cdots$$

$$k_1, k_2, k_3, k_4, k_5, \cdots = 01001\cdots$$
$$c_1, c_2, c_3, c_4, c_5, \cdots = 11111\cdots \tag{6-17}$$

插入一位 $p=1$，得到

$$p_1, p_2, p_3, p_4, p_5, \cdots = 110110$$
$$k_1, k_2, k_3, k_4, k_5, k_6, \cdots = 010010$$
$$c_1, c_2, c_3, c_4, c_5, c_6, \cdots = 100100 \tag{6-18}$$

这里只知道两个密文流以及 p，由上述分析得

$$p \oplus k_2 = c_2 \Rightarrow k_2 = p \oplus c_2 = 1$$
$$p_2 \oplus k_2 = c_2 \Rightarrow p_2 = k_2 \oplus c_2 = 0$$
$$p_2 \oplus k_3 = c_3 \Rightarrow k_3 = p_2 \oplus c_3 = 0$$

依次可以得到明文序列和密钥序列，当然这里只能得到插入位后的序列。

6.10　RC4 和 RC5

6.10.1　RC4

RC4 加密算法是大名鼎鼎的 RSA 三人组中的头号人物 Ron Rivest 在 1987 年设计的密钥长度可变的流加密算法簇。之所以称其为簇，是由于其核心部分的 S-box 长度可为任意，但一般为 256 字节。该算法的速度可以达到 DES 加密的 10 倍左右。

RC4 算法本身很简单，对于 n 位长的字，它总共有 $n=2^n$ 个可能的内部置换状态矢量 $\boldsymbol{S}$，这些状态是保密的。典型地 $n=8$，即以一个字节为单位，此时，用从 1 到 256 个字节（即 8 到 2048 位）的可变长度密钥初始化一个 256 个字节的状态数组 S。S 的元素记为 $S[0]$，$S[1]$，…，$S[255]$，自始至终置换后的 $\boldsymbol{S}$ 包含从 0 到 255 的所有 8 比特数。密钥流 K 由 S 中 256 个元素按一定的方式选出一个元素生成，每生成一个 K 值，S 中的元素就被重新置换一次。

RC4 有两个主要的算法：密钥调度算法（KSA）和伪随机数生成算法（PRGA）。

KSA 开始初始化 S，即 $S(i)=i(i=0\sim255)$。通过选取一系列数字，并加载到密钥数组 $K(0)\sim K(255)$。不用去选取这 256 个数，只要不断重复直到 K 被填满。数组 S 可以利用以下程序来实现随机化：

```
j: = 0;
for i: = 0 to 255 do begin
  j: = i + S(i) + K(i)(mod 256)
  swap (S(i),S(j))
end
```

一旦 KSA 完成了 S 的初始随机化，PRGA 就将接手工作，它为密钥流选取字节，即从 S 中选取随机元素，并修改 S 以便下一次选取。选取过程取决于索引 i 和 j，这两个索引值都是从 0 开始的。下面的程序就是选取密钥流的每个字节：

```
i: = i + 1(mod 256)
```

```
j: = j + S(i)(mod 256)
swap(S(i),S(j))
  t: = S(i) + S(j)(mod 256)
k: = S(t)
```

以 3 位(0～7)的 RC4 为例，其操作是对 8 取模，而不是 256。数组 S 有 8 个元素，如果选取 5、6 和 7 作为密钥，利用循环构建实际的 S 数组：

```
j = 0;
for i = o to 7 do
j = (j + S(i) + K(i)) mod 8;
swap (S(i),S(j));
```

该循环以 $j=0$ 和 $i=0$ 开始，使用更新公示后 j 为

$$j = [0 + S(0) + K(0)] \bmod 8 = (0 + 0 + 5) \bmod 8 = 5$$

因此，S 数组的第一个操作是将 $S(0)$ 与 $S(5)$ 互换。

索引 i 加 1 后，j 的下一个值为

$$j = [5 + S(1) + K(1)] \bmod 8 = (5 + 1 + 6) \bmod 8 = 4$$

因此，S 数组的 $S(1)$ 与 $S(4)$ 互换。

当循环执行后，数组 S 被随机化用来生成随机数序列。从 $j=0$ 和 $i=0$ 开始，RC4 按照下述计算第一个随机数：

$$i = (i + 1) \bmod 8 = (0 + 1) \bmod 8 = 1$$

$$j = [(j + S(i))] \bmod 8 = [(0 + S(1))] \bmod 8 = (0 + 4) \bmod 8 = 4$$

$$\text{swap}S(1)\text{and}S(4)$$

$$t = [S(j) + S(i)] \bmod 8 = [S(4) + S(1)] \bmod 8 = (1 + 4) \bmod 8 = 5$$

$$k = S(t) = S(5) = 6$$

第一个随机数为 6，其二进制表示为 110。反复进行该过程，直到生成的二进制位等于明文的位。

常见的 RC4 实现基于 $n=8$，这种系统的初始密钥是 0～256 的整数，共有 2^{1600} 种可能。这相当于使用了一个 1600 位的密钥，使强力攻击法变得不可能。

6.10.2 RC5

RC5 分组密码算法是在 RC4 的基础上进行改进的，它是参数可变的分组密码算法，三个可变的参数是：分组大小、密钥大小和加密轮数。在此算法中使用了三种运算：异或、加和循环。

RC5 是种比较新的算法，Rivest 设计了 RC5 的一种特殊的实现方式，因此 RC5 算法有一个面向字的结构：RC5-$w/r/b$，这里 w 是字长，其值可以是 16、32 或 64。对于不同的字长，明文和密文块的分组长度为 $2w$ 位，r 是加密轮数，b 是密钥字节长度。由于 RC5 一个分组长度可变的密码算法，为了便于说明在本文中主要是针对 64 位的分组 $w=32$ 进行处理的，下面详细说明了 RC5 加密解密的处理过程。

1. 创建密钥组

RC5 算法加密时使用了 $2r+2$ 个密钥相关的 32 位字，这里 r 表示加密的轮数。创建这

个密钥组的过程是非常复杂的但也是直接的，首先将密钥字节拷贝到 32 位字的数组 L 中(此时要注意处理器是 little-endian 顺序还是 big-endian 顺序)，如果需要，最后一个字可以用零填充。然后利用线性同余发生器模 2 初始化数组 S。

对于 $i=1$ 到 $2(r+1)-1$(本应模，本文中令 $w=32$)：

其中对于 16 位字 32 位分组的 RC5，$P=$0xb7e1，$Q=$0x9e37。

对于 32 位字和 64 位分组的 RC5，$P=$0xb7e15163，$Q=$0x9e3779b9。

对于 64 位字和 128 位分组，$P=$0xb7151628aed2a6b，$Q=$0x9e3779b97f4a7c15。

最后将 L 与 S 混合，混合过程如下：

$i=j=0$

$A=B=0$

处理 $3n$ 次(这里 n 是 $2(r+1)$和 c 中的最大值，其中 c 表示输入的密钥字的个数)。

2. 加密处理

在创建完密钥组后开始进行对明文的加密，加密时，首先将明文分组划分为两个 32 位字：A 和 B(在假设处理器字节顺序是 little-endian、$w=32$ 的情况下，第一个明文字节进入 A 的最低字节，第四个明文字节进入 A 的最高字节，第五个明文字节进入 B 的最低字节，以此类推)，其中操作符<<<表示循环左移，加运算是模(本应模，本文中令 $w=32$)的。

输出的密文是在寄存器 A 和 B 中的内容。

3. 解密处理

解密也是很容易的，把密文分组划分为两个字：A 和 B(存储方式和加密一样)，这里符合>>>是循环右移，减运算也是模(本应模，本文中令 $w=32$)的。

RSA 试验室花费了相当的时间来分析 64 位分组的 RC5 算法，在 5 轮后统计特性看起来非常好。在 8 轮后，每一个明文位至少影响一个循环。对于 5 轮的 RC5，差分攻击需要 2^{24}个选择明文；对 10 轮需要 2^{45}个；对于 12 轮需要 2^{53}个；对 15 轮需要 2^{68}个。而对于 64 位的分组只有 2^{64}个可能的明文，所以对于 15 轮或以上的 RC5 的差分攻击是失败的。在 6 轮后线性分析就是安全的了，Rivest 推荐至少 12 轮，甚至可能是 16 轮。这个轮数可以进行选择。

解密函数定义如下：

```
void RC5_Block_Decrypt (RC5_WORD * S, int R, char * in, char * out)
{
int i;
RC5_WORD A,B;
A = in[0] & 0xFF;
A  +=  (in[1] & 0xFF) << 8;
A  +=  (in[2] & 0xFF) << 16;
A  +=  (in[3] & 0xFF) << 24;
B = in[4] & 0xFF;
B  +=  (in[5] & 0xFF) << 8;
B  +=  (in[6] & 0xFF) << 16;
B  +=  (in[7] & 0xFF) << 24;
```

```
for(i = R;i >= 1;i -- ){
B = ROTR((B - S[2 * i + 1]),A,W);
B = B ^A;
A = ROTR((A - S[2 * i]),B,W);
A = A ^B;
}
B = B - S[1];
A = A - S[0];
out[0] = (A >> 0) & 0xFF;
out[1] = (A >> 8) & 0xFF;
out[2] = (A >> 16) & 0xFF;
out[3] = (A >> 24) & 0xFF;
out[4] = (B >> 0) & 0xFF;
out[5] = (B >> 8) & 0xFF;
out[6] = (B >> 16) & 0xFF;
out[7] = (B >> 24) & 0xFF;
return;
}/ * End of RC5_Block_Decrypt * /
int RC5_CBC_Decrypt_Init (pAlg,pKey)
rc5CBCAlg * pAlg;
rc5UserKey * pKey;
{
if ((pAlg == ((rc5CBCAlg * ) 0)) ||
(pKey == ((rc5UserKey * ) 0)))
return (0);
RC5_Key_Expand (pKey -> keyLength,pKey -> keyBytes,pAlg -> R,pAlg -> S);
return (RC5_CBC_SetIV(pAlg,pAlg -> I));
}
int RC5_CBC_Decrypt_Update(rc5CBCAlg * pAlg,int N,char * C,int * plainLen,char * P)
{
int plainIndex,cipherIndex,j;
plainIndex = cipherIndex = 0;
for(j = 0;j < BB;j++)
{
P[plainIndex] = pAlg -> chainBlock[j];
plainIndex++;
}
plainIndex = 0;
while(cipherIndex < N)
{
if(pAlg -> inputBlockIndex < BB)
{
pAlg -> inputBlock[pAlg -> inputBlockIndex] = C[cipherIndex];
pAlg -> inputBlockIndex++;
cipherIndex++;
}
if(pAlg -> inputBlockIndex == BB)
{
pAlg -> inputBlockIndex = 0;
RC5_Block_Decrypt (pAlg -> S,pAlg -> R,pAlg -> inputBlock,pAlg -> chainBlock);
for(j = 0;j < BB;j++)
```

```
{
if(plainIndex < BB)
P[plainIndex]^ = pAlg -> chainBlock[j];
else
P[plainIndex] = C[cipherIndex - 16 + j]^pAlg -> chainBlock[j];
plainIndex++;
}
}
}
*plainLen = plainIndex;
return (1);
}/* End of RC5_CBC_Decrypt_Update */
```

习题

1．同步序列密码和自同步序列密码的区别是什么？

2．序列密码和分组密码的区别是什么？

3．判断序列 110101100100011 的随机性。

4．图 6-5 所示是一个 3-级反馈移位寄存器，反馈函数 $f(x)=a_3 \oplus a_1$，初态为 110，求其输出序列的前 8 位。

5．利用图 6-13 的加密体制加密明文(m)字母 B，10-级反馈移位寄存器，其中参数及初态分别为$(c_1c_2\cdots c_{n-1}c_n)=(1000111001)$，$(a_na_{n-1}\cdots a_2a_1)=(0011101001)$，求密文($e$)。

第7章 数字签名

数字签名由公钥密码发展而来，它在网络安全，包括身份认证、数据完整性、不可否认性以及匿名性等方面有着重要应用。

7.1 数字签名概述

7.1.1 数字签名的产生

信息安全所面临的基本攻击类型，包括被动攻击（获取消息的内容、业务流分析）和主动攻击（假冒、重放、消息的篡改、业务拒绝）。抗击被动攻击的方法是加密，抗击主动攻击的方法是消息认证。

消息认证是一个过程，用以验证接收消息的真实性（的确是由它所声称的实体发来的）和完整性（未被篡改、插入、删除），同时还用于验证消息的顺序性和时间性（未重排、重放、延迟）。

报文认证用以保护双方之间的数据交换不被第三方侵犯，但它并不保证双方自身的相互欺骗。假定 A 发送一个认证的信息给 B，双方之间的争议可能有多种形式：

B 伪造一个不同的消息，但声称是从 A 收到的；

A 可以否认发过该消息，B 无法证明 A 确实发了该消息。

为了进一步确认双方的真实性，数字签名应运而生，数字签名是认证的重要工具。

数字签名不是指将你的签名扫描成数字图像，或者用触摸板获取的签名，更不是你的落款。经过数字签名的文件的完整性是很容易验证的（不需要骑缝章，骑缝签名，也不需要笔迹专家），而且数字签名具有不可抵赖性（不需要笔迹专家来验证）。

简单地说，所谓数字签名就是附加在数据单元上的一些数据，或是对数据单元所做的密码变换。这种数据或变换允许数据单元的接收者用以确认数据单元的来源和数据单元的完整性并保护数据，防止被人（例如接收者）伪造。它是对电子形式的消息进行签名的一种方法，一个签名消息能在一个通信网络中传输。

数字签名与消息认证的区别主要体现在：

数字签名：第三者可以确认收发双方的消息传送。

消息认证：只有收发双方才能确认消息的传送。

数字签名与手工签名的区别主要体现在：

数字签名：数字的，因消息而异。

手工签名：模拟的，因人而异。

7.1.2 数字签名的原理

1. 数字签名应具有的性质

(1) 收方能确认或证实发方的签字，但不能伪造。

(2) 发方发出签名后的消息，就不能否认所签的消息。

(3) 收方对已收到的消息不能否认。

(4) 第三者可以确认收发双方之间的消息传送，但不能伪造这一过程。

(5) 必须能够验证签名者及其签名的日期时间。

(6) 必须能够认证被签名消息的内容。

(7) 签名必须能够由第三方验证，以解决争议。

2. 数字签名应满足的要求

(1) 签字的产生必须使用发方独有的一些信息以防伪造和否认。

(2) 签字的产生应较为容易。

(3) 签字的识别和验证应较为容易。

(4) 对已知的数字签名构造一新的消息或对已知的消息构造一假冒的数字签名在计算上都是不可行的。

3. 数字签名的流程

数字签名的流程如图 7-1 所示，具体的签名步骤为：

(1) 发送方将明文信息通过 HASH 函数变成消息摘要。

(2) 将消息摘要用私钥加密。

(3) 将加密后的摘要连同明文一起发送给接收方。

(4) 接收方也将明文信息通过 HASH 函数变成消息摘要。

(5) 将加密后的摘要用发送方的公钥进行解密。

(6) 比较解密的信息是否与消息摘要吻合，如果吻合即成功签名。

4. 数字签名的执行方式

数字签名的执行方式有两类：直接方式和具有仲裁的方式。直接数字签名仅涉及通信双方，有效性依赖发方密钥的安全性；仲裁数字签名使用第三方认证。

直接方式是指数字签名的执行过程只有通信双方参与，并假定双方有共享的秘钥或接收一方知道发方的公钥。

直接数字签名的缺点是验证模式依赖于发送方的保密密钥，发送方要抵赖发送某一消息时，可能会声称其私有密钥丢失或被窃，从而他人伪造了他的签名。通常需要采用与私有密钥安全性相关的行政管理控制手段来制止或至少是削弱这种情况，但威胁在某种程度上依然存在。

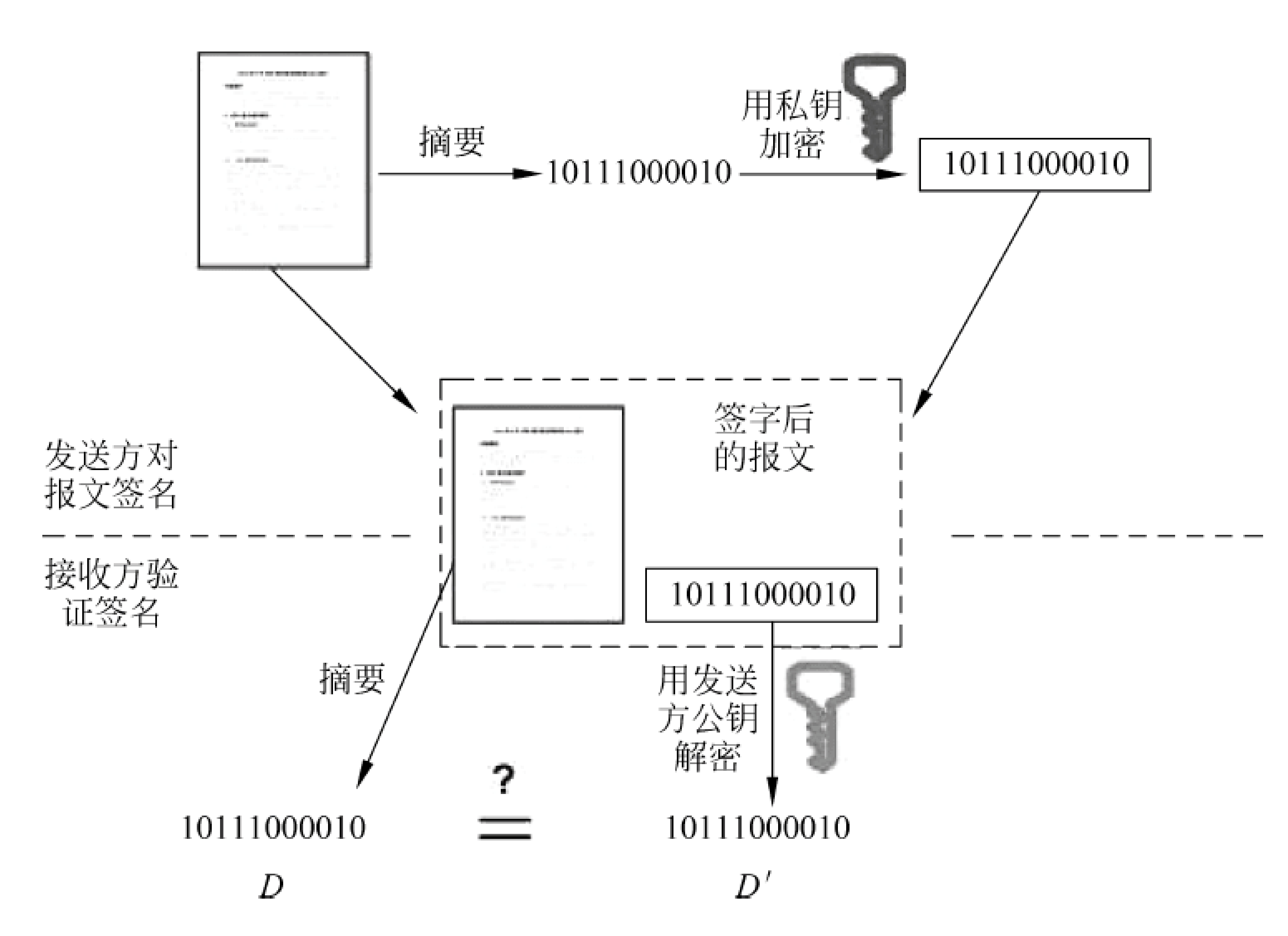


图 7-1 数字签名的流程

改进的方法可以要求被签名的信息包含一个时间戳(日期与时间),并要求将已暴露的密钥报告给一个授权中心。例如 X 的私有密钥确实在时间 T 被窃取,敌方可以伪造 X 的签名并附上早于或等于时间 T 的时间戳。

引入仲裁者,通常的做法是所有从发送方 X 到接收方 Y 的签名消息首先送到仲裁者 A,A 将消息及其签名进行一系列测试,以检查其来源和内容,然后将消息加上日期并与已被仲裁者验证通过的指示一起发给 Y。

仲裁者在这一类签名模式中扮演敏感和关键的角色,所有的参与者必须极大地相信这一仲裁机制工作正常。

7.2 利用 RSA 公钥密码体制实现数字签名

利用 RSA 公钥密码体制可以很容易地实现数字签名。

设 A 是发送方,B 是接收方,M 为明文,$K_{eA}=\langle e,n\rangle$是 A 的公开密钥,$K_{dA}=\langle d,p,q,\varphi(n)\rangle$是 A 的保密密钥,则 A 对 M 的签名过程为

$$S_A = D(M,K_{dA}) = (M^d) \bmod n$$

验证签名的过程为

$$E(S_A,K_{eA}) = (M^d)^e \bmod n = M$$

如果要同时确保数据的秘密性和真实性,则可以采用先签名后加密的方案,即

(1) A 对 M 签名:$S_A=D(M,K_{dA})$。

(2) A 对签名加密:$E(S_A,K_{eB})$。

(3) A 将 $E(S_A,K_{eB})$发送给 B。

RSA 的数字签名很简单,但实际应用还要注意很多问题。

1. 一般攻击

由于 RSA 密码的加密运算和解密运算具有相同的形式，都是模幂运算。设 e 和 n 是用户 A 的公开密钥，所以任何人都可以获得并使用 e 和 n。攻击者首先随意选择一个数据 Y，并用 A 的公开密钥计算 $X=(Y)^e \bmod n$，于是便可以用 Y 伪造 A 的签名。因为 X 是 A 对 Y 的一个有效签名。

这种攻击实际上成功率是不高的，因为对于随意选择的 Y，通过加密运算后得到的 X 具有正确语义的概率是很小的。可以通过认真设计数据格式或采用 HASH 函数与数字签名相结合的方法阻止这种攻击。

2. 利用已有的签名进行攻击

假设攻击者想要伪造 A 对 M_3 的签名，可以很容易地找到另外两个数据 M_1 和 M_2，使得

$$M_3 = M_1M_2 \bmod n$$

首先设法让 A 分别对 M_1 和 M_2 进行签名：

$$S_1 = (M_1)^d \bmod n$$

$$S_2 = (M_2)^d \bmod n$$

这时攻击者就可以用 S_1 和 S_2 计算出 A 对 M_3 的签名 S_3：

$$(S_1S_2) \bmod n = ((M_1)^d(M_2)^d) \bmod n = ((M_3)^d) \bmod n = S_3$$

对付这种攻击的方法是用户不要轻易对其他人提供的随机数据进行签名。更有效的方法是不直接对数据签名，而是对数据的 HASH 值签名。

3. 利用签名进行攻击获得明文

如果攻击者截获了密文 C，$C=M^e \bmod n$，想要求出明文 M，可以通过选择一个小的随机数 r，计算

$$x = r^e \bmod n$$

$$y = xC \bmod n$$

$$t = r^{-1} \bmod n$$

因为 $x=r^e \bmod n$，所以

$$x^d = (r^e)^d \bmod n \Rightarrow x^d = r \bmod n$$

可以让发送者对 y 签名，于是攻击者又获得

$$S = y^d \bmod n$$

计算

$$tS \bmod n = r^{-1}y^d \bmod n = r^{-1}x^dC^d \bmod n = C^d \bmod n = M$$

于是明文 M 可求。

对付这种攻击的方法也是用户不要轻易地对其他人提供的随机数据进行签名。最好是不直接对数据签名，而是对数据的 HASH 值签名。

4. 对先加密后签名方案的攻击

假设用户 A 采用先加密后签名的方案把 M 发送给用户 B，则先用 B 的公开密钥 e_B 对

M 加密，然后用自己的私钥 d_A 签名。再设 A 的模为 n_A，B 的模为 n_B。于是 A 发送数据给 B：

$$((M)^{e_B} \bmod n_B)^{d_A} \bmod n_A$$

如果 B 是不诚实的，则他可以用 M_1 抵赖 M，而 A 无法争辩。因为 n_B 是 B 对模，所以 B 知道 n_B 的因子分解，于是就能计算模 n_B 的离散对数。然后可以公布新公开的密钥为 xe_B。这时就可以宣布收到的是 M_1 而不是 M。

A 无法争辩的原因在于下式成立：

$$((M_1)^{xe_B} \bmod n_B)^{d_A} \bmod n_A = ((M)^{e_B} \bmod n_B)^{d_A} \bmod n_A$$

为了对付这种攻击，发送者应当在发送的数据中加入时间戳，从而证明是用 e_B 对 M 加密，而不是用新公开的密钥 xe_B 对 M_1 加密的。另一个对付这种攻击的方法是经过 HASH 处理后再签名。

总之，对付对数字签名攻击最好的方法是不要直接对数据签名，而应对数据的 HASH 值签名；其次是要采用先签名后加密的数字签名方案，而不是采用先加密后签名的数字签名方案。

7.3 数字签名标准

数字签名标准(Digital Signature Standard，DSS)是 1991 年 8 月由美国 NIST 公布的，1994 年 5 月 19 日正式公布，并于 1994 年 12 月 1 日被采纳为美国联邦信息处理标准。DSS 为 EIGamal 和 Schnorr 签名方案的改进，其使用的算法记为 DSA(Digital Signature Algorithm)，此算法由 D. W. Kravitz 设计。DSS 使用了 SHA，安全性基于求离散对数的困难性。

7.3.1 DSS 的基本方式

RSA 算法既能用于加密和签字，又能用于密钥交换。与此不同，DSS 使用的算法只能提供数字签名功能。图 7-2 用于比较 RSA 签字和 DSS 签字的不同方式。

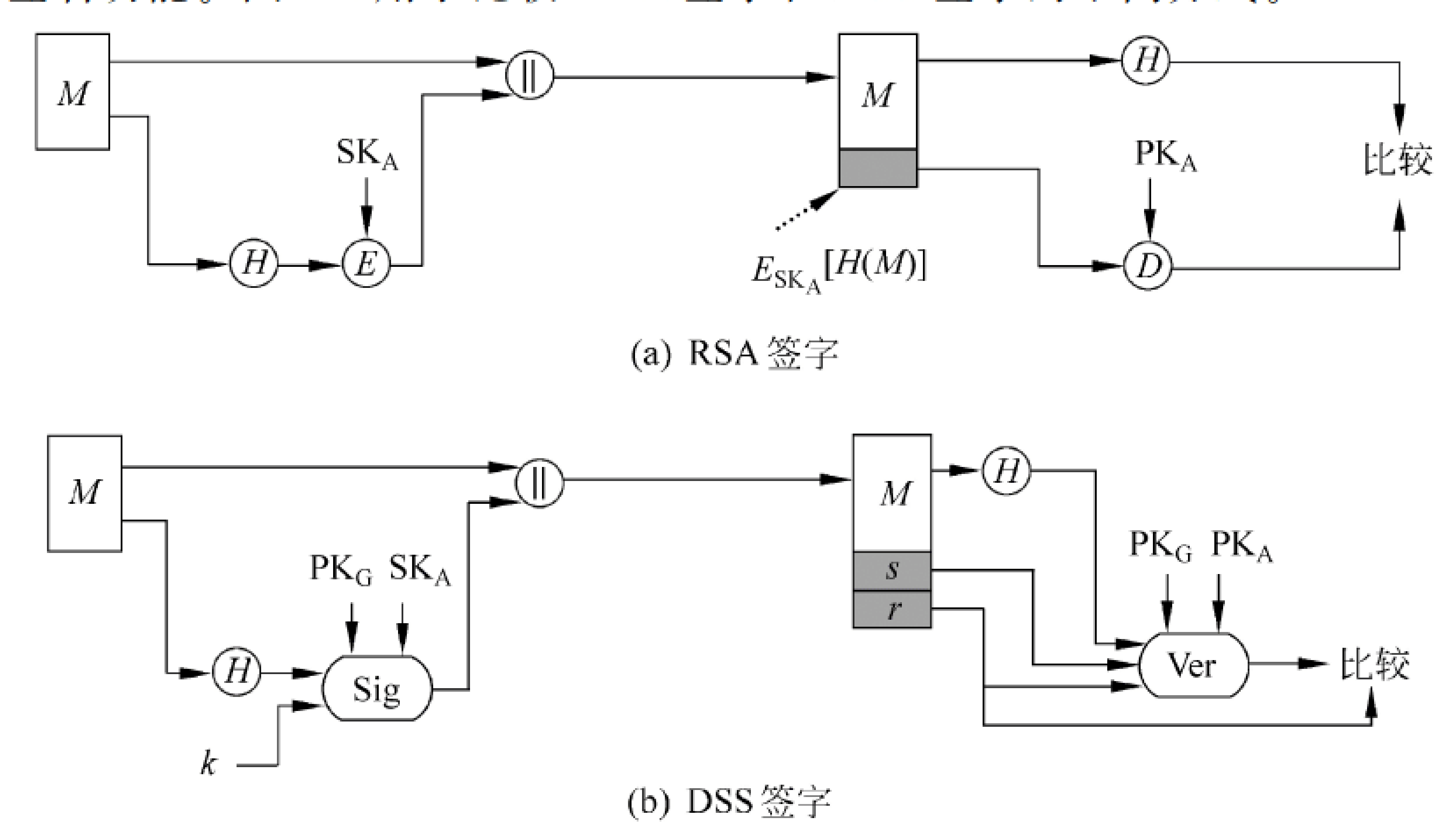


图 7-2 RSA 签字与 DSS 签字的不同方式

采用 RSA 签字时，将消息输入一个杂凑函数以产生一个固定长度的安全杂凑值，再用发方的密钥加密杂凑值就形成了对消息的签字。消息及其签字被一起发给收方，收方得到消息后再产生消息的杂凑值，且使用发方的公钥对收到的签字解密。这样收方就得到了两个杂凑值，如果两个杂凑值是一样的，则认为收到的签字是有效的。

DSS 签字也利用一杂凑函数产生消息的一个杂凑值，杂凑值连同一随机数 k 一起作为签字函数的输入，签字函数还需使用发送方的密钥 SK_A 和供所有用户使用的一组参数，称这一组参数为全局公钥 PK_G。签字函数的两个输出 s 和 r 就构成了消息的签字 (s,r)。接收方收到消息后再产生消息的杂凑值，将杂凑值与收到的签字一起输入验证函数，验证函数还需输入全局公钥 PK_G 和发送方的公钥 PK_A。验证函数的输出如果与收到的签字成分 r 相等，则验证了签字是有效的。

7.3.2 DSA 算法

DSA 算法的签字过程和验证过程如图 7-3 所示，其具体步骤为：

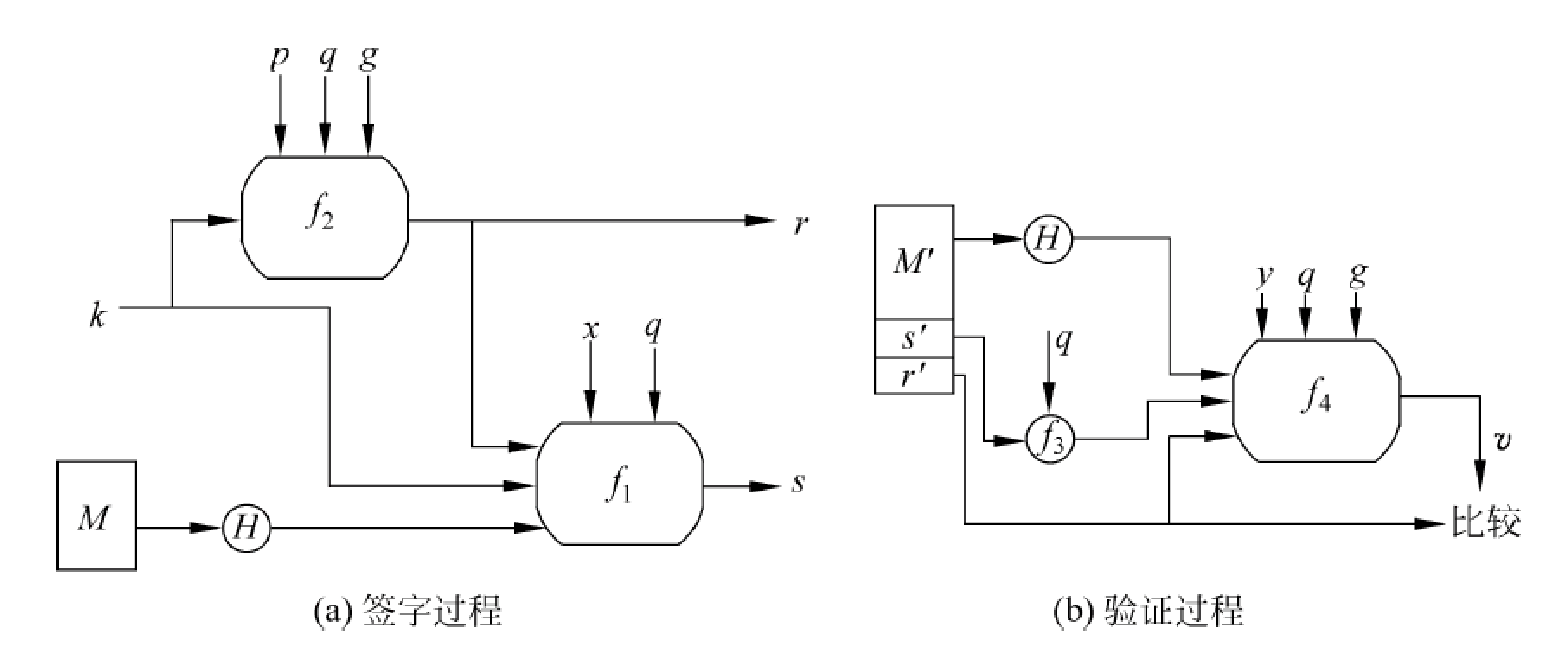


图 7-3 DSA 算法的签字过程和验证过程

(1) 公开参数。

p：512 至 1024 位的大质数；

q：160 位的 $p-1$ 之质因子；

g：$g=w^{(p-1)/q} \bmod p$，其中 $w<p-1$ 且 $w^{(p-1)/q} \bmod p>1$；

h：一个单向杂凑函数，输出值为 160 位。

注： 搭配 DSA 的单向杂凑函数标准为 SHA-1。

(2) 密钥产生。每一个使用者任选一个整数 $x\in Z_q$ 为私钥，并计算公钥 $y=g^x \bmod p$。

(3) 签署程序。

任选一数 $k<q$。

计算 $r=(g^k \bmod p) \bmod q$。

计算 $s=k^{-1}(h(M)+xr) \bmod q$。

(r,s) 为 M 的数字签章。

(4) 验证程序。

计算

$$a = s^{-1} \bmod q$$

$$b = ah(M) \bmod q$$
$$c = ra \bmod q$$
$$d = (g^b \times y_c \bmod p) \bmod q$$

若 $d=r$，则(r,s)通过验证。

7.4 其他签字方案

7.4.1 GOST 数字签名算法

GOST 数字签名算法是俄罗斯于1995年公布的数字签名标准，其做法与DSA很类似，但具有使用弹性，安全度也较高，其具体的步骤为：

(1) 算法参数。

p：509～512位或1020～1024位的一个大质数。

q：254～256位的 $p-1$ 之质因子。

g：任意小于 $p-1$ 的整数，满足 $g^q \bmod p=1$。

h：一个单向杂凑函数。

GOST 数字签名算法使用一个单向HASH函数。

(2) 密钥产生。每一个使用者任选一个整数 $x \in Z_q$ 为私钥，并计算公钥 $y=g^x \bmod p$。

(3) 签署程序。

任选一数 $k<q$。

计算

$$r = (g^k \bmod p) \bmod q$$
$$s = (xr + kh(M)) \bmod q$$

若 $h(M) \bmod q=0$，则令 $h(m)=1$；

若 $r=0$，则重复回步骤(1)执行。

(r,s)为 M 的数字签名。

(4) 验证程序。

计算以下参数：

$$a = h(M)^{q-2} \bmod q$$
$$b = sa \bmod q$$
$$c = a(q-r) \bmod q$$
$$d = (g^b \times c^c \bmod p) \bmod q$$

若 $d=r$，则(r,s)通过验证。

理论上GOST签名与ELGamal签名和DSS类似。为了更加安全，GOST签名采用了更大的素数 q。一般学者认为160位的 q 就足够了，因此DSS采用160位，但GOST签名却采用了256位的 q，这说明俄罗斯希望自己的签名标准更安全。由于参数 q 较大，故其签名速度比DSS慢。在DSS中采用的签名公式为 $s=(sr+k^{-1}(\text{SHA}(M)) \bmod q)$，而在GOST签名中却采用了 $s=((sr+kH(M)) \bmod q)$，其区别为一个使用参数 k，一个使用参数 k^{-1}，

从而导致签名的验证过程也不同。

7.4.2 不可否认的数字签名算法

不可否认的数字签名是1989年由Chaum和Antwerpen提出的。不可否认的数字签名是一种特殊的数字签名，它具有一些新颖的特征，没有签名者的合作，接收者就无法验证签名，在某种程度上保护了签名者的利益。

不可否认的数字签名有一些特定的应用场合，例如软件开发者可利用不可否认的数字签名对他们的软件进行保护，使得只有付了钱的顾客才能验证签名并相信开发者仍然对软件负责。

一个不可否认的签名的真伪性是通过接收者和签名者执行一个协议来推断的，这个协议称做否认协议。因为签名者可声称一个合法的签名是伪造的，在这种情况下，如果签名者拒绝参加验证，就可以认为签名者有欺骗行为。如果签名者参加验证，由否认协议就可推断出签名者的真伪性。

一个不可否认的数字签名方案由三部分组成：签名算法、验证协议和否认协议。签名算法和验证的步骤为：

(1) 系统公开参数。

p：大质数；

g：mod p 的一个本原元。

(2) 密钥产生。

私钥：$x \in Z_p$；

公钥：$y = g^x \bmod p$。

(3) 签署程序。签名：$x = m^x \bmod p$(欲签署信息为 m)。

(4) 验证签名程序。

验证者任选二数 $a, b < p$，计算 $c = z^a (g^x)^b \bmod p$，并将 c 送给原签署者。

原签署者计算 $d = c^x \bmod p$，并将之送给验证者。

验证者检验 $d = m^a g^b \bmod p$ 是否成立。若成立，则 z 通过验证。

7.4.3 Fail-Stop数字签名算法

Fail-Stop数字签名的不可伪造性依赖于一个计算假设，但是如果一个签名被伪造，那么假定的签名者能证明这个签名是一个伪造签名。更精确地说，能证明做基础的计算假设一定被攻破，这个证明伪造的能力不依赖于任何密码假设并独立于伪造者的计算能力，这样能保护一个多项式界定签名者免遭具有无限计算能力的伪造者的攻击。再者，在第一次伪造之后，系统的所有参加者或系统操作人员都知道签名方案已被攻破，因此系统将终止工作，这也是这个系统为什么称做Fail-Stop(失败-停止)的原因。

Fail-Stop数字签名方案适合于联用对电子支付系统，在这个系统中可使顾客无条件地安全，顾客无需担心银行能攻破签名方案的基础假设。

一个Fail-Stop数字签名方案主要由三个算法组成：签名算法、验证算法和伪造证明。一个安全的Fail-Stop数字签名方案应具有以下特性：

(1) 如果签名者正确地签一个消息,那么接收者接收这个签名;

(2) 一个多项式界定 DF 伪造者不能构造签名使之通过验证;

(3) 如果一个具有无限计算能力的伪造者成功地构造了一个签名使之通过验证,那么这个概率是极小的,签名者能产生一个伪造证明使得第三方相信一个伪造已经发生;

(4) 一个多项式界的签名者不能构造一个假签名使后来证明是一个伪造。

达到这些特性的基本观点是对应于每个公钥有许多密钥,并且对同一个消息不同的密钥给出不同的签名。签名者恰好知道这些密钥中的一个,并且对一个给定的消息只能构造可能签名中的一个。然而,对一个新消息即使一个具有无限计算能力的伪造者也没有充分的消息确定签名者能构造众多可能的签名中的哪一个。因此,一个伪造签名以很高的概率不同于签名者已经构造的签名。而对同一个消息的两个不同签名的知识产生了一个伪造证明。

7.4.4 基于离散对数问题的数字签名法

基于离散对数问题的数字签名法是以 ElGamal 方法为主要代表的,其具体的步骤为

(1) 公开参数。

p:一个大质数;

q:为 $p-1$ 或 $p-1$ 的一个大质因子;

g:$1<g<q$,满足 $g^q=1 \bmod p$;

h:一个单向杂凑函数。

(2) 密钥产生。每一个使用者任选一个整数 $x\in Z_q$ 为私钥,并计算公钥 $y=g^x \bmod p$。

(3) 签署及验证签章。签署一个信息 m 时(通常 m 需经过 h 转换以防止选择密文攻击),签署者首先任选一个随机数 $k\in Z_q$,满足 $\gcd(k,q)=1$;接下来,计算 $r=g^k \bmod p$ 与 $r'=r \bmod q$。

数字签章(r,s)产生与验证公式如表 7-1 所示。

表 7-1 签章产生与验证式

签章产生式	签章验证式
$r'k=s+mx(\bmod q)$	$r^{r'}=g^s\times y^m(\bmod p)$
$r'k=m+sx(\bmod q)$	$r^{r'}=g^m\times y^s(\bmod p)$
$sk=r'+mx(\bmod q)$	$r^s=g^{r'}\times y^m(\bmod p)$
$sk=m+r'x(\bmod q)$	$r^s=g^m\times y^{r'}(\bmod p)$
$mk=s+r'x(\bmod q)$	$r^m=g^s\times y^{r'}(\bmod p)$
$mk=r'+sx(\bmod q)$	$r^m=g^{r'}\times y^s(\bmod p)$

7.4.5 Ong-Schnorr-Shamir 签章法

此签章法是由 Ong. Schnorr 与 Shamir 于 1984 年所提出的,但在 1987 年被证明为不安全,直到 1993 年才有人提出一个安全的改进方法。该签章法最大的优点为计算复杂度相当低,大部分用于计算能力较差的 IC 卡或行动通信系统的应用,其具体的步骤为:

(1) 密钥产生。首先,系统公布一个大合成数 n,但秘密保留 n 所含的大质因子 p 与 q。

任选一数 k，满足 $\gcd(k,n)=1$，并计算

$e=-k^{-2} \bmod n=-(k^{-1})^2 \bmod n$。

公钥为 (e,n)，私钥为 k。

(2) 签署程序(欲签署信息为 m)。

任选一数 r，满足 $\gcd(r,n)=1$。

计算

$$s_1=\frac{1}{2}\times(mr^{-1}+r) \bmod n$$

$$s_2=\frac{k}{2}\times(mr^{-1}-r) \bmod n$$

(s_1,s_2) 为 m 的数字签章。

(3) 验证程序。

签章验证式：$s_1^2+e\times s_2^2=m(\bmod n)$。

7.4.6 ESIGN 签章法

ESIGN 签章法是由日本 NTT 的 T. Okamoto 于 1990 年所发明的方法，可以视为日本的数字签名标准。1999 年，ESIGN 也正式被列入 ISO 国际标准。在相同的密钥长度与签章大小的条件之下，ESIGN 签署与验证程序比 RSA 或 DSA 都要来得快速，其具体的步骤为：

(1) 密钥产生。任选一组大质数 p 与 q(至少为 192 位)，并计算 $n=p^2\times q$，其中，n 为公钥，p 与 q 为私钥。

公布一个单向杂凑函数 h 与一个整数 k，其中 $h(m)$ 的输出值，介于 0 与 $m-1$ 之间，$k\in\{8,16,32,64,128,256,512,1024\}$。

(2) 签署程序(欲签署信息为 m)。任选一数 k(视安全需求而定)。

计算

$$w=\left\lceil\frac{(h(m)-x^k) \bmod n}{p\times q}\right\rceil$$

$$s=x+(w\times(k\times x^{k-1})^{-1} \bmod p)\times p\times q$$

s 为 m 的数字签章。

(3) 验证程序。

若 $h(m)\leqslant s^k \bmod n<h(m)+2^{2|n|/3}$，则 s 通过验证，$|n|$ 为 n 的位数。

7.4.7 盲签名算法

在普通数字签名中，签名者总是先知道数据的内容后才实施签名，这是通常的办公事务所需要的。但有时却需要某个人对某数据签名，而又不能让他知道数据的内容，称这种签名为盲签名。在无记名投票选举和数字化货币系统中往往需要这种盲签名，因此盲签名在电子商务和电子政务系统中有着广泛的应用前景。

盲签名与普通签名相比有两个显著的特点：

(1) 签名者不知道所签署的具体内容。

(2) 在签名被接收者泄露后，签名者不能追踪签名。

为了满足这两个条件，接收者首先将待签数据进行盲变换，把变换后的盲数据发给签名者，经签名者签名后再发给接收者。接收者对签名再进行去盲变换，得出的便是签名者对原数据的盲签名。这样便满足了条件1。要满足条件2，必须使签名者事后看到盲签名时不能与盲数据联系起来，这通常是依靠某种协议来实现的。

盲签名的具体步骤为：

(1) 系统参数。如同RSA所定义的参数，签署者的公钥为(e,n)，私钥为d。

(2) 签署程序。假设A欲让B签署一个信息m，但不让B知道m。

A任选一随机数k，$1<k<n$，并计算$t=m\times k^e \bmod n$，随后，A将t送给B签署。

B签署t，亦即$t^d=(m\times k^e)^d \bmod n$。

A计算m的签名如下：$s=t^d\times k^{-1} \bmod n$，亦即$s=m^d \bmod n$。

(3) 验证程序。

$$m = S^e \bmod n$$

7.4.8 代理签名算法

代理签名(Agent Signature Scheme)是指用户由于某种原因指定某个代理代替自己签名。例如，A处长需要出差，而这些地方不能很好地访问计算机网络。因此A希望接收一些重要的电子邮件，并指示其秘书B做相应的回信。A在不把其私钥给B的情况下，可以请B代理，这种代理具有下面的特性：

(1) 任何人都可区别代理签名和正常的签名。

(2) 不可伪造性：只有原始签名者和指定的代理签名者能够产生有效的代理签名，代理签名者必须创建一个能检测到是代理签名的有效代理签名。

(3) 可验证性：从代理签名中，验证者能够相信原始的签名者认同了这份签名消息。

(4) 可识别性：原始签名者能够从代理签名中识别代理签名者的身份。

(5) 不可否认性：代理签名者不能否认由他建立且被认可的代理签名。

代理签名的具体步骤为：

(1) 系统公开参数。

p：大质数；

g：$\bmod p$的一个本原元。

(2) 原始签署者的私钥及公钥(x,y)：$x\in Z_p$，$y=g^x \bmod p$。

(3) 代理密钥产生程序：原始签署者执行以下步骤。

任选一随机数$k\in Z_{p-1}$，计算$K=g^k \bmod p$与$\sigma=x+kK \bmod p-1$。

将代理密钥(σ,K)秘密传送给代理签署者。

(4) 代理验证程序。

代理签署者检验代理密钥的有效性：$g^\sigma=yK^K \bmod p$。

若上式成立，则代理签署者接受(σ,K)为有效代理密钥；否则退回(σ,K)并要求原始签署者另外代理密钥，或终止执行以下程序。

(5) 签署程序。令m为欲签署信息。代理签署者使用σ作为签署密钥，利用基于离散

对数 mod p 的数字签名技术产生签名 $\mathrm{Sig}_\sigma(m)$，并将$(m,\mathrm{Sig}_\sigma(m),K)$传送给验证者。

(6) 验证签名程序。验证者计 $y'=yK^K \bmod p$，将 y'视为原始签署者新的公钥，并执行签名验证程序以验证代理签名。

7.5 认证协议

安全可靠的通信除需进行消息的认证外，还需建立一些规范的协议对数据来源的可靠性、通信实体的真实性加以认证，以防止欺骗、伪装等攻击。

网络通信的一个基本问题：A 和 B 是网络的两个用户，他们想通过网络先建立安全的共享密钥再进行保密通信。那么 A(B)如何确信自己正在和 B(A)通信而不是和 C 通信呢？这种通信方式为双向通信，因此，此时的认证称为相互认证。对于单向通信来说，认证称为单向认证。

相互认证是最常用的协议，该协议使得通信各方互相认证鉴别各自的身份，然后交换会话密钥。基于认证的密钥交换的核心问题包括保密性和时效性。

为了防止伪装和暴露会话密钥，基本认证与会话密码信息必须以保密形式通信。这就要求预先存在保密或公开密钥以实现加密使用，同时要防止消息重放攻击。

所谓消息重放是指在最坏情况下可能导致向敌人暴露会话密钥，或成功地冒充其他人，至少也可以干扰系统的正常运行，处理不好将导致系统瘫痪。常见的消息重放攻击形式有：

(1) 简单重放。攻击者简单复制一条消息，以后再重新发送它。

(2) 可被日志记录的重放。攻击者可以在一个合法有效的时间窗内重放一个带时间戳的消息。

(3) 不能被检测到的重放。这种情况可能出现，原因是原始信息已经被拦截，无法到达目的地，而只有重放的信息到达目的地。

(4) 反向重放，不做修改。向消息发送者重放。当采用传统对称加密方式时，这种攻击是可能的。因为消息发送者不能简单地识别发送的消息和收到的消息在内容上的区别。

对付重放攻击的一种方法是在认证交换中使用一个序列号来给每一个消息报文编号。仅当收到的消息序数顺序合法时才接受之。但这种方法的困难是要求双方必须保存上次消息的序号。

保证消息的实时性常用的有时间戳和询问-应答两种方法。

时间戳：如果 A 收到的消息包括一时间戳，且在 A 看来这一时间戳充分接近自己的当前时刻，A 才认为收到的消息是新的并接受之。这种方案要求所有各方的时钟是同步的。

询问-应答：用户 A 向 B 发出一个一次性随机数作为询问，如果收到 B 发来的消息(应答)也包含一个正确的一次性随机数，A 就认为 B 发来的消息是新的并接受之。

时间戳方法似乎不能用于面向连接的应用，因为该技术固有的困难包括：

(1) 某些协议需要在各种处理器时钟中维持同步。该协议必须既要容错以对付网络出错，又要安全以对付重放攻击；

(2) 由于某一方的时钟机制故障可能导致临时失去同步，这将增大攻击成功的机会；

(3) 由于变化的和不可预见的网络延迟的本性，不能期望分布式时钟保持精确的同步。

因此，任何基于时间戳的过程必须采用时间窗的方式来处理：一方面时间窗应足够大以包容网络延迟，另一方面时间窗应足够小以最大限度地减小遭受攻击的机会。

而询问-应答方式则不适合无连接的应用过程，这是因为在无连接传输以前需经询问-应答这一额外的握手过程，这与无连接应用过程的本质特性不符。对无连接的应用程序来说，利用某种安全的时间服务器保持各方时钟同步是防止重放攻击最好的方法。

7.6 散列函数

7.6.1 单向散列函数

散列函数(Hash Function)是一种从任何一种数据中创建小的数字“指纹”的方法。该函数将数据打乱混合，重新创建一个叫做散列值的指纹。散列值通常用来代表一个短的随机字母和数字组成的字符串。好的散列函数在输入域中很少出现散列冲突。在散列表和数据处理中，不抑制冲突来区别数据，会使得数据库记录更难找到。

所有散列函数都有一个基本特性：如果两个散列值是不相同的(根据同一函数)，那么这两个散列值的原始输入也是不相同的。这个特性使散列函数具有确定性的结果。但另一方面，散列函数的输入和输出不是唯一对应的关系，如果两个散列值相同，两个输入值很可能是相同的。但也可能不同，这种情况称为杂凑碰撞，这通常是两个不同长度的散列值，刻意计算出相同的输出值。输入一些数据计算出散列值，然后部分改变输入值，一个具有强混淆特性的散列函数会产生一个完全不同的散列值。

由于散列函数应用的多样性，它们经常是专为某一应用而设计的。例如，加密散列函数假设存在一个要找到具有相同散列值的原始输入的敌人。一个设计优秀的加密散列函数是一个单向操作：对于给定的散列值，没有实用的方法可以计算出一个原始输入，也就是说很难伪造。以加密散列为目的设计的函数，如 MD5，被广泛地用作检验散列函数。这样软件下载的时候，就会对照验证代码之后才下载正确的文件部分。此代码有可能因为环境因素的变化，如机器配置或者 IP 地址的改变而有变动，以保证源文件的安全性。错误监测和修复函数主要用于辨别数据被随机的过程所扰乱的事例。当散列函数被用于校验和的时候，可以用相对较短的散列值来验证任意长度的数据是否被更改过。

定义 7-1 如果函数 h 满足下列性质，则称 h 是一个散列函数。

(1) 压缩性：任意有限长度的输入 x，为固定长度的输出 $h(x)$；

(2) 易计算：给定输入 x，$h(x)$是易计算的。

显然，将消息对应到散列函数值是一个多对一定映射，所以，可能出现多个不同消息具有相同散列函数值的情形，这种现象称为碰撞。

7.6.2 无碰撞散列函数和离散对数散列函数

当攻击者对一个散列函数进行攻击时，一种可能的方式是：设(x,y)是一个有效的签名，$h(x)$是消息 x 的散列函数值，$y=\mathrm{sig}_k(h(x))$是对散列值 $h(x)$的签名。如果攻击者找到一个消息 $x'\neq x$，$h(x')=h(x)$，则(x',y)是一个有效签名，但它是一个伪造签名。当然，一

般地，x'只能是一个随机消息，不过，攻击者达到了干扰通信的目的。为了阻止这种攻击，散列函数必须满足下面定义的弱无碰撞性。

定义 7-2 如果对给定的一个消息 x，找到一个满足 $x' \neq x, h(x') = h(x)$的消息 x 是计算上不可能的，则称散列函数 h 是弱无碰撞的。

另一种可能的攻击是：攻击者事先找到两个消息 $x' \neq x, h(x') = h(x)$，并骗得签名者对消息 x 的签名，则$(x', h(x')) = (x', h(x))$就是一个合法的伪造签名。

为了防止这种伪造，需要更强的无碰撞性。

定义 7-3 如果找到两个满足 $x' \neq x, h(x') = h(x)$的消息 x, x'是计算上不可能的，则称散列函数 h 是强无碰撞的。

显然，如果一个散列函数是强无碰撞的，则该函数一定是弱无碰撞的。

可以证明强无碰撞性包含了单向性。事实上，如果一个散列函数不具有单向性，而具有一个逆算法，则存在一个寻找碰撞的概率算法 Las Vagas 算法，能找到一个碰撞的概率至少为$\frac{1}{2}$。

可见，强无碰撞性是一个很强的概念，它不但包含了弱无碰撞性，又包含了单向性。所以，安全的散列函数应该是强无碰撞的。

Chaum-van Heijst-Pfitzmann 提出一个基于离散对数的散列函数，其描述如下：

假设 $p = 2q + 1$ 是一个大素数，其中 q 也是一个素数。设 α 和β 是 Z_p^* 中的两个生成元，离散对数 $\log_\alpha \beta$ 的计算是困难的，将 α 和 β 保密。

散列函数

$$h: Z_q \times Z_q \to Z_p^*$$

定义为

$$h(x_1, x_2) = \alpha^{x_1} \beta^{x_2} \bmod p$$

下面的定理表示只要离散对数是安全的，则这个散列函数是安全的，即散列函数是强无碰撞的。

定理 7-1 如果能找到 Chaum-van Heijst-Pfitzmann 散列函数 h 的一个碰撞，则离散对数 $\log_\alpha \beta$ 可有效计算。

签名讨论的散列函数具有有限的定义域，显然不能完全满足需要。当要签名文本文件时，被签名消息可以是任意长度的。所以，需要具有无限定义域的强无碰撞散列函数。将具有有限定义域的强无碰撞散列函数扩展为具有无限定义域的强无碰撞散列函数通常的方法是：通过级联的方式来构造无限长度输入的散列函数，以及 Merkle(Merkle-Damgard)级联算法，或者是基于分组加密算法的散列函数扩展。

7.6.3 单向散列函数的设计

单向散列函数通常有以下技术要求：

（1）单向散列函数能够处理任意长度的明文（至少是在实际应用中可能碰到的长度的明文），其生成的消息摘要数据块长度具有固定的大小。而且，对同一个消息反复执行该函数总是得到相同的信息摘要。

（2）单向散列函数生成的信息摘要是不可预见的，消息摘要看起来和原始的数据没有

任何关系。而且，原始数据的任何微小变化都会对生成的信息摘要产生很大的影响。

(3) 具有不可逆性，即通过生成的报文摘要得到原始数据的任何信息在计算上是完全不可行的。

(4) 由散列值寻找等价明文的困难性。给定 M，要找到另一消息 M'，并满足 $H(M)=H(M')$ 是计算上不可行的，即弱碰撞。

(5) 寻找等价明文对的困难性。找到两个不同的消息 M 和 M'，使它们的散列值相等即 $H(M)=H(M')$，在计算上不可行的，即强碰撞(强碰撞比弱碰撞要求更加严格，它可以有效地抵抗所谓的“生日攻击”)。

给定输入 m，单向散列函数首先将 m 分成若干个分组($m_1, m_2, \cdots, m_k$)，并以迭代的方式，从第 1 个分组开始，每次处理一个分组。

在处理第 i 个分组时，将分组 i 与处理分组 $i-1$ 时得到的散列值作为压缩函数 f 的输入，即

$$h_i = f(m_i, h_{i-1})$$

这时压缩函数的输出将作为直到分组 i 的输入的散列值。最后，在处理最后一个分组时得到的散列值，将作为整个报文的散列值输出。

在设计时要求单向散列函数具有强的码间相关性，即修改明文中的一个比特，就会使输出比特串中大约一半的比特发生变化。这样，最后得到的散列值将与明文的每一个比特密切相关。

单向散列函数的原理比较简单，同时由于它并不要求可逆，因此，设计自由度一般比较大，基本的设计方法有三种。

1. 使用公开密钥密码算法

通常可以以 CBC 模式使用公开密钥算法对消息进行加密，并输出最后一个密文分组作为散列值。

如果丢弃用户的保密密钥，这时的散列值将无法解密，也就是说，它满足了散列函数的单向性要求。

虽然在合理的假设下，可以证明这类散列函数是安全的，但一般情况下它的计算速度十分慢(这是由公开密钥算法的速度决定的)。这一类散列函数并不实用。

2. 使用对称分组算法

使用对称分组密码算法的 CBC 模式或 CFB 模式来产生散列值。它将使用一个固定的密钥加密消息，并将最后的密文分组作为散列值输出。这时，如果分组算法是安全的，那么单向函数也将是安全的。

另外，还可以把消息作为密钥，而把上一个分组得到的散列值作为分组算法的输入，使用类似于 CBC 模式的方法进行加密，最后得到的密文分组作为散列值输出。假设消息共有 N 个分组 $m_1, m_2, \cdots, m_N$，用 h_i 表示到第 i 个分组时的散列值，h 表示最后的散列输出。那么，可以表述如下：

$$h_0 = \mathrm{IV}$$
$$h_i = E_{m_i}(h_{i-1})$$

$$h = h_N$$

其中,IV 是初始向量。

这类设计已经提出一些方案,如 Quisquater-Girault 算法、MDC-2 和 MDC-4、GOST 散列函数等。

3. 直接设计单向散列函数

这类单向散列函数并不基于任何假设和密码体制,它通过直接构造复杂的非线性关系达到单向性要求。

这类算法典型的有 MD2、MD4、MD5、SHA-1、PIPE-MD 和 HAVAL 等算法。

目前,直接设计单向散列函数的方法受到了广泛关注,是比较流行的一种设计方法

7.6.4 单向散列函数的安全性

对单向散列函数的攻击是指攻击者寻找一对产生碰撞消息的过程。评价单向散列函数最好的方法就是看一个攻击者找到一对碰撞消息所花的代价有多高。通常在讨论单向散列函数的安全时,都假设敌手已经知道单向散列函数算法(遵循 Kerckhoffs 假设)。

对单向散列函数的攻击方法可以分为两类:

第一类是强力攻击,它可以用于对任何类型的单向散列函数进行攻击,其中典型的方法称为"生日攻击"。

生日攻击方法没有利用 HASH 函数的结构和任何代数弱性质,它只依赖于消息摘要的长度,即 HASH 值的长度。这种攻击对 HASH 函数提出了一个必要的安全条件,即消息摘要必须足够长。生日攻击这个术语来自于所谓的生日问题,在一个教室中最少应有多少学生才使得至少有两个学生的生日在同一天的概率不小于 1/2? 这个问题的答案为 23。

采用生日攻击的攻击者将产生许多的报文,计算其报文摘要,并进行比较。当报文足够多时,根据概率论的有关结论,报文将以某种较大的概率发生碰撞,这时可以认为散列函数已经被攻破。这种攻击可行性的关键在于究竟需要多少报文才能使发生碰撞的概率足够大。如果报文数目大到是计算上不可行的,那么生日攻击就是不可行的,否则,可以认为该单向散列函数是不安全的。

生日攻击所需报文数目与摘要值的长度有关。摘要值越长,需要的报文数目越大。有计算表明,当摘要长度为 n 时,需要 $2\,\frac{n}{2}$个报文就可以以 50%的概率产生一个碰撞。

强碰撞自由性质指出计算发生碰撞的报文是计算上不可行的。因此,生日攻击对具有强碰撞自由性质的单向散列函数无法奏效。

第二类攻击方法依赖于单向散列函数的结构和代数性质,它采用针对单向散列函数弱性质的方法进行攻击。这一类攻击方法有中间相遇攻击、修正分组攻击和差分分析等。另外,使用了其他密码算法构造的单向散列函数还可以因为所使用的密码算法的弱点而引起攻击。例如,DES 的一些众所周知的弱点,如互补性、弱密钥与半弱密钥等,都可用来攻击基于 DES 构造的单向散列函数。

7.7 MD5

MD5(Message Digest Algorithm),中文名为消息摘要算法第五版,为计算机安全领域广泛使用的一种散列函数,用以提供消息的完整性保护。

Rivest 在 1989 年开发出 MD2 算法。在这个算法中,首先对信息进行数据补位,使信息的字节长度是 16 的倍数。然后,以一个 16 位的检验和追加到信息末尾,并且根据这个新产生的信息计算出散列值。后来,Rogier 和 Chauvaud 发现如果忽略了检验将和 MD2 产生冲突。

为了加强算法的安全性,Rivest 在 1990 年又开发出 MD4 算法。MD4 算法同样需要填补信息以确保信息的比特位长度加上 448 后能被 512 整除(信息比特位长度 mod 512=448)。然后,一个以 64 位二进制表示的信息的最初长度被添加进来。Den boer 和 Bosselaers 以及其他人很快发现了攻击 MD4 版本中第一步和第三步的漏洞。毫无疑问,MD4 就此被淘汰了。尽管 MD4 算法在安全上有个这么大的漏洞,但它对在其后才被开发出来的好几种信息安全加密算法的出现却有着不可忽视的引导作用。

一年以后,即 1991 年,Rivest 开发出技术上更趋近成熟的 MD5 算法。它在 MD4 的基础上增加了“安全-带子”(safety-belts)的概念。虽然 MD5 比 MD4 稍微慢一些,但却更为安全。这个算法很明显地由 4 个和 MD4 设计有少许不同的步骤组成。在 MD5 算法中,信息-摘要的大小和填充的必要条件与 MD4 完全相同。

MD5 的作用是让大容量信息在用数字签名软件签署私人密钥前被“压缩”成一种保密的格式(就是把一个任意长度的字节串变换成一定长的大整数)。不管是 MD2、MD4 还是 MD5,它们都需要获得一个随机长度的信息并产生一个 128 位的信息摘要。MD5 最广泛地被用于各种软件的密码认证和钥匙识别上。通俗地说就是人们所说的序列号。

MD5 接受任意长度的消息作为输入,并生成 128 位消息摘要作为输出。对于给定的长度为 L 位的消息,建立算法需要三个步骤。

(1) 通过在消息末尾添加一些额外位来填充消息。填充是绝大多数散列函数的通用特性,正确地填充能够提高算法的安全性。对于 MD5 来说,对消息进行填充,使其位长度等于 448 mod 512(这是小于 512 位一个整数倍的 64 位)。即使原始消息达到了所要求的长度,也要添加填充。填充由一个 1 和足够个数的 0 组成,以便达到所要求的长度。例如,如果消息由 704 位组成,那么在其末尾要添加 256 位(1 后面跟 255 个 0),以便将消息扩展到 960 位(960 mod 512=448)。

(2) 消息的原始长度缩减为 mod 64,然后以一个 64 位的数字添加到扩展后消息的尾部。在这个示例中,原始消息的长度为 704 位,其二进制值为 1011000000。将这个数书写为 64 位数字(在开始位置添加 54 个 0),并把它添加到消息末尾。其结果是一个具有 1024 位的消息。

(3) MD5 的初始输出放在 4 个 32 位寄存器 A、B、C 和 D 中,这些寄存器随后将用于保存散列函数的中间结果和最终结果。初始值为(十六进制)

$$A=67452301;\quad B=\text{EFCDAB89};\quad C=\text{98BADCFE};\quad D=10325476$$

一旦完成了这些步骤,MD5 将以四轮方式处理每一个 512 位块。每一轮都由 16 个阶

段组成，都实现针对该轮的功能，对消息块部分做 32 位加法，对数组中的内置值做 32 位加法，移位运算，最后做一次加法和交换运算。从而真正地打乱了所有位。

以一个例子来解析 MD5 的具体加密过程：

如对一个字符串 string 进行加密。第一步，要把它转换成位(MD5 是对位进行操作的)，现在假设 string 转换为位后是 1010000101110101。第二步就要将这个字节串补位成比 512 的倍数(n 倍)位少 64 位，补位的规则就是在原来位后先补一个 1，其他的补 0，补位后这个串变成 1010000101110101 1(先补的那个 1)0000...000(总共(512－64)位)。完成这些后还要在其后面补上一个 64 位的数据，当然这个数据也是有规定的，这个数据就是原字节串的长度(当然这个长度已被转换成了 64 位)。至此，数据补完后这串正好是 512 的倍数：$512N-64+64$。

至此，前两步补位和补数据长度完成了，在一些初始化处理后，MD5 以 512 位分组来处理输入文本，每一分组又划分为 16 个 32 位子分组。算法的输出由 4 个 32 位分组组成，将它们级联形成一个 128 位散列值。首先填充消息，使其长度恰好为一个比 512 位的倍数仅小 64 位的数。填充方法是附一个 1 在消息后面，后接所要求的多个 0，然后在其后附上 64 位的消息长度(填充前)。这两步的作用是使消息长度恰好是 512 位的整数倍(算法的其余部分要求如此)，同时确保不同的消息在填充后不相同。

4 个 32 位变量初始化为

$A=$0x01234567

$B=$0x89abcdef

$C=$0xfedcba98

$D=$0x76543210

它们称为链接变量(Chaining Variable)，接着进行算法的主循环，循环的次数是消息中 512 位消息分组的数目。

将上面 4 个变量复制到另外的变量中：A 到 a，B 到 b，C 到 c，D 到 d。

主循环有 4 轮(MD4 只有 3 轮)，每轮很相似。第一轮进行 16 次操作。每次操作对 a、b、c 和 d 中的其中三个做一次非线性函数运算，然后将所得结果加上第四个变量，文本的一个子分组和一个常数。再将所得结果向右循环移一个不定的数，并加上 a、b、c 或 d 中之一。

最后用该结果取代 a、b、c 或 d 中之一。以下是每次操作中用到的 4 个非线性函数(每轮一个)。

$F(X,Y,Z)=(X\&Y)|((\sim X)\&Z)$

$G(X,Y,Z)=(X\&Z)|(Y\&(\sim Z))$

$H(X,Y,Z)=X\,\hat{}\,Y\,\hat{}\,Z$

$I(X,Y,Z)=Y\,\hat{}\,(X|(\sim Z))$

(& 是与，| 是或，～是非，^是异或)

这些函数是这样设计的：如果 X、Y 和 Z 的对应位是独立和均匀的，那么结果的每一位也应是独立和均匀的。

函数 F 是按逐位方式操作的：如果 X，那么 Y，否则 Z。函数 H 是逐位奇偶操作符。设 Mj 表示消息的第 j 个子分组(从 0 到 15)，$<<<s$ 表示循环左移 s 位，则 4 种操作为

FF(a,b,c,d,Mj,s,ti)表示 $a=b+(a+(F(b,c,d)+Mj+ti)<<<s)$

GG(a,b,c,d,Mj,s,ti)表示 $a=b+(a+(G(b,c,d)+Mj+ti)<<<s)$

HH(a,b,c,d,Mj,s,ti)表示 $a=b+(a+(H(b,c,d)+Mj+ti)<<<s)$

II(a,b,c,d,Mj,s,ti)表示 $a=b+(a+(I(b,c,d)+Mj+ti)<<<s)$

这四轮(64 步)如下。

第一轮：

FF(a,b,c,d,M0,7,0xd76aa478)
FF(d,a,b,c,M1,12,0xe8c7b756)
FF(c,d,a,b,M2,17,0x242070db)
FF(b,c,d,a,M3,22,0xc1bdceee)
FF(a,b,c,d,M4,7,0xf57c0faf)
FF(d,a,b,c,M5,12,0x4787c62a)
FF(c,d,a,b,M6,17,0xa8304613)
FF(b,c,d,a,M7,22,0xfd469501)
FF(a,b,c,d,M8,7,0x698098d8)
FF(d,a,b,c,M9,12,0x8b44f7af)
FF(c,d,a,b,M10,17,0xffff5bb1)
FF(b,c,d,a,M11,22,0x895cd7be)
FF(a,b,c,d,M12,7,0x6b901122)
FF(d,a,b,c,M13,12,0xfd987193)
FF(c,d,a,b,M14,17,0xa679438e)
FF(b,c,d,a,M15,22,0x49b40821)

第二轮：

GG(a,b,c,d,M1,5,0xf61e2562)
GG(d,a,b,c,M6,9,0xc040b340)
GG(c,d,a,b,M11,14,0x265e5a51)
GG(b,c,d,a,M0,20,0xe9b6c7aa)
GG(a,b,c,d,M5,5,0xd62f105d)
GG(d,a,b,c,M10,9,0x02441453)
GG(c,d,a,b,M15,14,0xd8a1e681)
GG(b,c,d,a,M4,20,0xe7d3fbc8)
GG(a,b,c,d,M9,5,0x21e1cde6)
GG(d,a,b,c,M14,9,0xc33707d6)
GG(c,d,a,b,M3,14,0xf4d50d87)
GG(b,c,d,a,M8,20,0x455a14ed)
GG(a,b,c,d,M13,5,0xa9e3e905)
GG(d,a,b,c,M2,9,0xfcefa3f8)
GG(c,d,a,b,M7,14,0x676f02d9)
GG(b,c,d,a,M12,20,0x8d2a4c8a)

第三轮：

HH(a,b,c,d,$M5$,4,0xfffa3942)
HH(d,a,b,c,$M8$,11,0x8771f681)
HH(c,d,a,b,$M11$,16,0x6d9d6122)
HH(b,c,d,a,$M14$,23,0xfde5380c)
HH(a,b,c,d,$M1$,4,0xa4beea44)
HH(d,a,b,c,$M4$,11,0x4bdecfa9)
HH(c,d,a,b,$M7$,16,0xf6bb4b60)
HH(b,c,d,a,$M10$,23,0xbebfbc70)
HH(a,b,c,d,$M13$,4,0x289b7ec6)
HH(d,a,b,c,$M0$,11,0xeaa127fa)
HH(c,d,a,b,$M3$,16,0xd4ef3085)
HH(b,c,d,a,$M6$,23,0x04881d05)
HH(a,b,c,d,$M9$,4,0xd9d4d039)
HH(d,a,b,c,$M12$,11,0xe6db99e5)
HH(c,d,a,b,$M15$,16,0x1fa27cf8)
HH(b,c,d,a,$M2$,23,0xc4ac5665)

第四轮：

II(a,b,c,d,$M0$,6,0xf4292244)
II(d,a,b,c,$M7$,10,0x432aff97)
II(c,d,a,b,$M14$,15,0xab9423a7)
II(b,c,d,a,$M5$,21,0xfc93a039)
II(a,b,c,d,$M12$,6,0x655b59c3)
II(d,a,b,c,$M3$,10,0x8f0ccc92)
II(c,d,a,b,$M10$,15,0xffeff47d)
II(b,c,d,a,$M1$,21,0x85845dd1)
II(a,b,c,d,$M8$,6,0x6fa87e4f)
II(d,a,b,c,$M15$,10,0xfe2ce6e0)
II(c,d,a,b,$M6$,15,0xa3014314)
II(b,c,d,a,$M13$,21,0x4e0811a1)
II(a,b,c,d,$M4$,6,0xf7537e82)
II(d,a,b,c,$M11$,10,0xbd3af235)
II(c,d,a,b,$M2$,15,0x2ad7d2bb)
II(b,c,d,a,$M9$,21,0xeb86d391)

常数 ti 可以选择如下：

在第 i 步中，ti 是 4 294 967 296×abs(sin(i))的整数部分，i 的单位是弧度。

所有这些完成之后，将 A、B、C、D 分别加上 a、b、c、d。然后用下一分组数据继续运行算法，最后的输出是 A、B、C 和 D 的级联。



习题

1. 数字签名应该具有哪些性质?

2. 数字签名应满足哪些要求?

3. 直接数字签名和仲裁数字签名的区别是什么?

4. 如果用于产生 DSA 签名的 k 已被泄密,会出现什么问题?

5. DSS 包括一个推荐的素数测试算法,该算法如下。

(1) 选择 w:令 w 是随机的奇数,则 $(w-1)$ 是偶数且可表示为 $2^a m$,其中 m 是奇数,也就是说,2^a 是整除 $(w-1)$ 的 2 的最大幂。

(2) 产生 b:令 b 是随机整数,$1<b<w$。

(3) 求幂:置 $j=0$,且 $x=b^m \bmod w$。

(4) 若 $j=0$,$x=1$ 或者 $x=w-1$,则 w 可能是素数,故应测试 w,转到步骤(8)。

(5) 若 $j>0$,$x=1$,则 w 不是素数,对该 w 算法终止。

(6) 置 $j=j+1$,若 $j<a$,则置 $z=x^2 \bmod w$,并转到步骤(4)。

(7) w 不是素数,对该 w 算法终止。

(8) 若已测试足够多的 b,则认为该 w 是素数并终止算法,否则转到步骤(2)。

请说明该算法的工作原理。

第8章 密钥管理

密钥管理是数据加密技术中的重要一环，密钥管理的目的是确保密钥的安全性（真实性和有效性）。

8.1 密钥管理技术的发展

第一代密钥管理产品是存储卡芯片钥匙。存储卡芯片钥匙是将存储卡芯片做成一个计算机外设，直接插在USB口上，密钥则写在存储卡芯片上。当存储卡芯片插在USB口上时，加密的文件可自动解密。当智能卡拔出时，文件便自动加密，而存储卡芯片内的密钥一般是不会被从计算机内读出来的，从而避免了密钥轻易被别人获取的可能。

社会的进步使得社会竞争进一步加剧，同时也产生了更多的商业机密。如何使自己的天机不被泄露，是每一个人都很关心的事。虽然第一代存储卡芯片钥匙能够解决部分问题，但仍然有其自身的缺陷，如使用读卡器即可读取智能卡内的文件。这个缺陷是致命的，也就是说别人可以轻松地配一把同样的钥匙，去打开计算机中加密的文件。

在此基础上，第二代密钥管理产品是安全钥匙。之所以称之为安全钥匙，主要是缘于其采用的是一款安全芯片。安全芯片，是指任何人采用任何暴力都无法读取安全芯片中的任何内容。这就是说，开启数据的钥匙是唯一的，这确保了计算机中加密保存的文件不会被其他任何人读取。这就好像，用户在计算机中创建一个保险箱，而钥匙永远只在自己手中。当把钥匙插在USB口上时，保险箱自动打开，可以把重要文件或应用程序放在里面，与计算机存盘一样轻松。钥匙拔出时，保险箱随之关闭，同时对文件加密。更进一步，除了加密外，它还将保险箱进行隐藏，当他人打开计算机时根本看不见在计算机中创建的保险箱。同时，由于对钥匙做了口令识别，从而确保钥匙的安全性。第二代安全钥匙的诞生，标志着安全加密的一个新时代的开始。

一个好的密钥管理系统应该做到：

（1）密钥难以被窃取；

（2）在一定条件下窃取了密钥也没有用，密钥有使用范围和时间的限制；

（3）密钥的分配和更换过程对用户透明，用户不一定要亲自掌管密钥。

8.2 密钥管理

8.2.1 密钥管理的内容

密钥的管理是整个加密系统中最薄弱的环节，密钥的泄漏将直接导致明文内容的泄漏。例如曾经有一种计算机使用了 DES 算法来实现一个文件加密工具，它将密钥与密文保存在一起，用户可以选择用密文或明文形式保存文件，而且加密/解密过程是透明的，使用很方便。但是，对于了解密文格式的攻击者而言，他可以很容易地发现密文的密钥，从而发现明文。显然从密钥管理的途径窃取机密比用破译的方法花费的代价要小得多，所以对密钥的管理和保护格外重要。

密钥管理包括管理方式、密钥生成、密钥存储和保护、密钥分配、传递和密钥备份、销毁等。所有管理过程都是为了正确地解决密钥从生成到使用全过程的安全性和实用性，另外还涉及密钥的行政管理制度和管理人员的素质。密钥管理最主要的过程是密钥生成、保护和分发。

1. 管理方式

层次化的密钥管理方式，用于数据加密的工作密钥需要动态产生；工作密钥由上层的加密密钥进行保护，最上层的密钥称为主密钥，是整个密钥管理系统的核心；多层密钥体制大大加强了密码系统的可靠性，因为用得最多的工作密钥常常更换，而高层密钥用得较少，使得破译者的难度增大。

2. 密钥的生成

密钥的生成与所使用的算法有关。如果生成的密钥强度不一致，则称该算法构成的是非线性密钥空间，否则称为线性密钥空间。

3. 分配、传递

密钥的分配是指产生并使使用者获得一个密钥的过程；密钥的传递分集中传送和分散传送两类。集中传送是指将密钥整体传送，这时需要使用主密钥来保护会话密钥的传递，并通过安全渠道传递主密钥。分散传送是指将密钥分解成多个部分，用秘密分享的方法传递，只要有部分到达就可以恢复，这种方法适用于在不安全的信道中传输。

4. 密钥的保存

密钥既可以作为一个整体保存，也可以分散保存。整体保存的方法有人工记忆、外部记忆装置、密钥恢复、系统内部保存；分散保存的目的是尽量降低由于某个保管人或保管装置问题而导致密钥的泄漏。

5. 备份、销毁

密钥的备份可以采用和密钥的分散保存一样的方式，以免知道密钥的人太多；密钥的

销毁要有管理和仲裁机制，否则密钥会被有意无意地丢失，从而造成对使用行为的否认。

8.2.2 密钥的组织结构

从信息安全的角度看，密钥的生存期越短，破译者的可乘之机就越小。所以，理论上一次一密最安全。在实际应用中，尤其是在网络环境下，多采用层次化的密钥管理结构。用于数据加密的工作密钥平时不存储于加密设备中，需要时动态生成，并由其上层的密钥加密密钥进行加密保护。密钥加密密钥可根据需要由其上一级的加密密钥进行保护。最高层的密钥被称为主密钥，它是整个密钥管理体系的核心。在多层密钥管理系统中，通常下一层的密钥由上一层密钥按照某种密钥算法来生成。因此，掌握了主密钥，就有可能找出下层的各个密钥。

工作密钥通常被称为会话密钥，建立会话密钥的目的在于：

(1) 重复使用密钥容易导致泄漏，因此应经常更换；

(2) 若使用相同的密钥，攻击者可将以前截获的信息插入当前的会话中而不被发现；

(3) 密钥一旦被破译，则使用这一密钥加密的信息都会失密，而使用会话密钥的会话信息也会失密；

(4) 如果对方不可靠，则更换会话密钥可防止对方以后窃取信息。

多层密钥管理体制大大增强了密码系统的安全性。由于用得最多的工作密钥经常更换，而高层密钥则用得较少，使得破译者可用的信息变得很少，增加了攻击的难度。

另外，多层密钥体制为自动化管理带来了方便，因为下层密钥可由计算机系统自动产生和维护，并通过网络自动分配和更换，减少了接触密钥的人数，也减轻了用户的负担。例如，在古典加密体制中有这样一种密钥管理方法：

(1) 指定一个公开出版并可广泛获得的出版物作为密码本，这时这个出版物的名称成为主密钥。

(2) 将这个出版物的某个页号 P、行号 L 及字数 W 作为第二级密钥。

(3) 将 P、L、W 指定的内容作为具体的密钥，即第三级密钥。

这样在使用时，主密钥是双方预知的，不需要交换。通信时，只要通知 P、L、W 就可得知加密的密钥，而破译者由于不知道主密钥，所以即使截获了密文和 P、L、W，也无法破译。如果指定的是一个连续出版物，则主密钥定期更换，它的期号或卷号成为新的一级密钥。这种超数学的密码结构使得密文、明文和密钥之间不存在任何确定的函数关系。破译者只能使用穷举法。当然破译者可通过分析加密者的生活习惯来缩小搜索范围。

密钥的连通是指在用户之间共享密钥的范围，而密钥的分割是指对这个范围的限制空间分割密钥，可区分不同的用户群，例如：

(1) 不同密级的数据之间的密钥分割；

(2) 不同业务部门、业务系统之间的密钥分割；

(3) 上下级机关之间的密钥分割；

(4) 应用系统和管理系统之间的密钥分割等。

按时间分割密钥可实现让各个用户在不同的时期使用不同的密钥，使用户的使用权具有时间限制。分割的实现有两种方式。

(1) 静态分割：在给用户的加密设备注入密钥时就给定了用户的密钥连通范围，即用

户只能使用注入的密钥；

（2）动态分割：密钥分配中心定期向规定范围内的用户加密传送一个用于控制分割范围的广播密钥（向指定用户广播的密钥）。

按照密钥的作用与类型及它们之间的相互控制关系，可以将不同类型的密钥划分为1级密钥、2级密钥、…、n级密钥，从而组成一个层密钥系统，如图8-1所示。

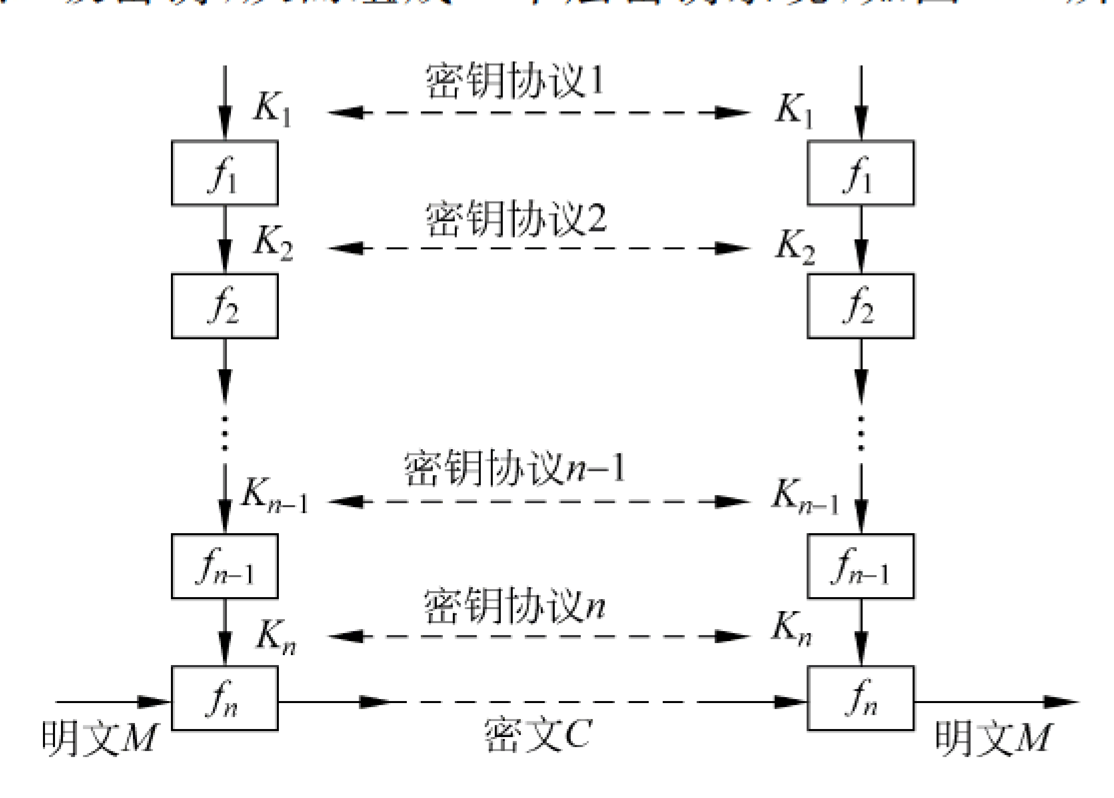


图8-1　层密钥系统

在图8-1中，系统使用一级密钥通过算法保护二级密钥（一级密钥使用物理方法或其他方法进行保护），使用二级密钥通过算法保护三级密钥，以此类推，直到最后使用级密钥通过算法保护明文数据。随着加密过程的进行，各层密钥的内容动态变化，而这种变化的规则由相应层次的密钥协议控制。

最下层的密钥也叫工作密钥，或数据加密密钥，它直接作用于对明文数据的加解密。所有上层密钥可称为密钥加密密钥，它们的作用是保护数据加密密钥或作为其他更低层次密钥的加密密钥。最上面一层的密钥也叫主密钥，通常主密钥是整个密钥管理系统的核心，应该采用最安全的方式来进行保护。

层次化的密钥结构意味着以密钥来保护密钥。这样，大量的数据可以通过少量动态产生的数据加密密钥（工作密钥）进行保护，而数据加密密钥又可以由更少量的、相对不变（使用期较长）的密钥加密密钥来保护。同理，在最后第二层的密钥加密密钥可以由主密钥进行保护，从而保证了除了主密钥可以以明文的形式存储在有严密物理保护的主机密码器件中，其他密钥则以加密后的密文形式存储，这样，就改善了密钥的安全性。

8.2.3 密钥的分配技术

密钥分配技术解决的是网络环境中需要进行安全通信的实体间建立共享的密钥问题，最简单的解决办法是生成密钥后通过安全的渠道送到对方。这对于密钥量不大的通信是合适的，但随着网络通信的不断增加，密钥量也随之增加，则密钥的传递与分配会成为严重的负担。在当前的实际应用中，用户之间的通信并没有安全的通信信道，因此有必要对密钥分配做进一步的研究。

密钥分配技术一般需要解决两个方面的问题：为减轻负担，提高效率，引入自动密钥分配机制；为提高安全性，尽可能减少系统中驻留的密钥量。

为了满足这两个问题，目前有两种类型的密钥分配方案：集中式和分布式密钥分配方

案。集中式密钥分配方案是指由密钥分配中心(KDC)或者由一组节点组成层次结构负责密钥的产生并分配给通信双方。分布式密钥分配方案是指网络通信中各个通信方具有相同的地位,它们之间的密钥分配取决于它们之间的协商,不受任何其他方的限制(更进一步,可以把密钥分配中心分散到所有的通信方,即每个通信方同时也是密钥分配中心)。

此外,密钥分配方案也可能采取上面两种方案的混合:上层(主机)采用分布式密钥分配方案,而上层对于终端或它所属的通信子网采用集中式密钥分配方案。

通信双方在使用对称密码技术进行保密通信时,通信双方必须有一个共享的密钥,并且这个密钥还要防止被他人获得。此外,密钥还必须时常更新。从这点上看,密钥分配技术直接影响密钥分配系统的强度。

对于通信双方 A 和 B,密钥分配可以有以下几种方法:

(1) 密钥由 A 选定,然后通过物理方法安全地传递给 B;

(2) 密钥由可信赖的第三方 C 选取并通过物理方法安全地发送给 A 和 B;

(3) 如果 A 和 B 事先已有一密钥,那么其中一方选取新密钥后,用已有的密钥加密新密钥发送给另一方;

(4) 如果 A 和 B 都有一个到可信赖的第三方 C 的保密信道,那么 C 就可以为 A 和 B 选取密钥后安全地发送给 A 和 B;

(5) 如果 A 和 B 都在可信赖的第三方 C 发布自己的公开密钥,那么他们用彼此的公开密钥进行保密通信。

对于前两种方法不适合大量连接的现代通信(因为需要对密钥进行人工传送);对于第(3)种方法,由于要对所有的用户分配初始密钥,代价也很大,也不适合于现代通信;对于第(4)种方法采用密钥分配技术,可信赖的第三方 C 就是密钥分配中心(KDC),常用于对称密码技术的密钥分配;对于第(5)种方法采用的是密钥认证中心技术,可信赖的第三方 C 就是证书授权中心(CA),常用于非对称密码技术的公钥分配。

8.3 PKI

8.3.1 PKI 综述

PKI 是 Public Key Infrastructure 的缩写,是指用公钥概念和技术来实施和提供安全服务的具有普适性的安全基础设施。这个定义涵盖的内容比较宽,是一个被很多人接受的概念。这个定义说明,任何以公钥技术为基础的安全基础设施都是 PKI。当然,没有好的非对称算法和好的密钥管理就不可能提供完善的安全服务,也就不能叫做 PKI。也就是说,该定义中已经隐含了必须具有的密钥管理功能。

X. 509 标准中,为了区别于权限管理基础设施(PMI),将 PKI 定义为支持公开密钥管理并能支持认证、加密、完整性和可追究性服务的基础设施。这个概念与第一个概念相比,不仅仅叙述 PKI 能提供的安全服务,更强调 PKI 必须支持公开密钥的管理。也就是说,仅仅使用公钥技术还不能叫做 PKI,还应该提供公开密钥的管理。因为 PMI 仅仅使用公钥技术但并不管理公开密钥,所以,PMI 就可以单独进行描述而不至于跟公钥证书等概念混淆。X.509 中从概念上分清 PKI 和 PMI 有利于标准的叙述。然而,由于 PMI 使用了公钥技术,

PMI 的使用和建立必须先有 PKI 的密钥管理支持。也就是说，PMI 不得不把自己与 PKI 绑定在一起。当把两者合二为一时，PMI＋PKI 就完全落在 X.509 标准定义的 PKI 范畴内。根据 X.509 的定义，PMI＋PKI 仍旧可以叫做 PKI，而 PMI 完全可以看成 PKI 的一个部分。

美国国家审计总署在 2001 年和 2003 年的报告中都把 PKI 定义为由硬件、软件、策略和人构成的系统，当完善实施后，能够为敏感通信和交易提供一套信息安全保障，包括保密性、完整性、真实性和不可否认。尽管这个定义没有提到公开密钥技术，但到目前为止，满足上述条件的也只有公钥技术构成的基础设施，也就是说，只有第一个定义符合这个 PKI 的定义。所以这个定义与第一个定义并不矛盾。

综上所述，可以认为：PKI 是用公钥概念和技术实施的，支持公开密钥的管理并提供真实性、保密性、完整性以及可追究性安全服务的具有普适性的安全基础设施。

8.3.2 PKI 的基本组成

完整的 PKI 系统必须具有权威认证机构（CA）、数字证书库、密钥备份及恢复系统、证书作废系统、应用接口（API）等基本构成部分，构建 PKI 也将围绕这五大系统来着手构建。

PKI 技术是信息安全技术的核心，也是电子商务的关键和基础技术。PKI 的基础技术包括加密、数字签名、数据完整性机制、数字信封、双重数字签名等。一个典型、完整、有效的 PKI 应用系统至少应具有以下部分：

（1）公钥密码证书管理。

（2）黑名单的发布和管理。

（3）密钥的备份和恢复。

（4）自动更新密钥。

（5）自动管理历史密钥。

（6）支持交叉认证。

认证机构（CA）：即数字证书的申请及签发机关，CA 必须具备权威性的特征。

数字证书库：用于存储已签发的数字证书及公钥，用户可由此获得所需的其他用户的证书及公钥。

密钥备份及恢复系统：如果用户丢失了用于解密数据的密钥，则数据将无法被解密，这将造成合法数据丢失。为避免这种情况，PKI 提供备份与恢复密钥的机制。但须注意，密钥的备份与恢复必须由可信的机构来完成。并且，密钥备份与恢复只能针对解密密钥，签名私钥为确保其唯一性而不能够作备份。

证书作废系统：证书作废处理系统是 PKI 的一个必备组件。与日常生活中的各种身份证件一样，证书有效期以内也可能需要作废，原因可能是密钥介质丢失或用户身份变更等。为实现这一点，PKI 必须提供作废证书的一系列机制。

应用接口（API）：PKI 的价值在于使用户能够方便地使用加密、数字签名等安全服务，因此一个完整的 PKI 必须提供良好的应用接口系统，使得各种各样的应用能够以安全、一致、可信的方式与 PKI 交互，确保安全网络环境的完整性和易用性。

通常来说，CA 是证书的签发机构，它是 PKI 的核心。众所周知，构建密码服务系统的核心内容是如何实现密钥管理。公钥体制涉及一对密钥（即私钥和公钥），私钥只由用户独

立掌握，无须在网上传输，而公钥则是公开的，需要在网上传送，故公钥体制的密钥管理主要是针对公钥的管理问题，目前较好的解决方案是数字证书机制。

由于 PKI 作为国家信息安全基础设施的重要战略地位及核心技术（密码技术）的特殊敏感性，中国 PKI 体系的发展与建立既不能简单地照搬国外的技术与构架，但是也不能盲目地完全走自由市场的道路。中国国家 PKI 体系应在国家控制和主导下，制定统一的发展战略规划和管理模式，由国家负责统一协调、管理和监控，打破一些行业内部的变相垄断，加强相关行业之间的合作，避免重复建设，促进平等竞争，建设一个有利于国家网络经济的体系，进而推动国民经济和社会信息化的发展。本着这样的原则，构建国家 PKI 体系的总体目标是：建设具有科学性、权威性、安全性和互通性的完整 PKI 体系，为国家信息化建设保驾护航。因此，为实现这样的目标，国家级 PKI 安全认证体系主要应由组织体系、管理体系、技术体系、标准体系和法律体系组成。结合已有的 PKI 认证体系，国家级 PKI 安全认证体系结构如图 8-2 所示。

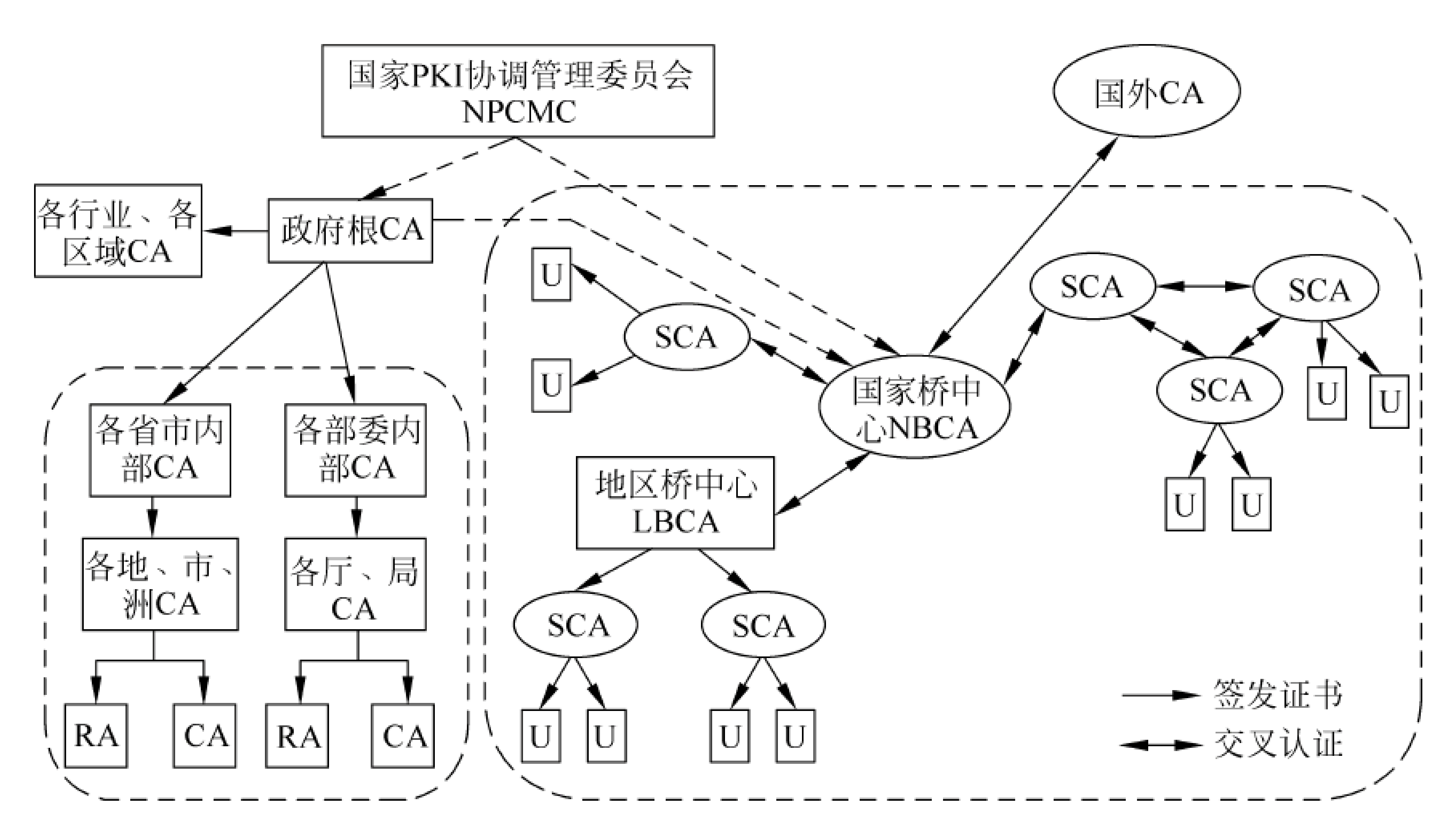


图 8-2 国家级 PKI 安全认证体系结构

1. 国家 PKI 协调管理委员会

作为组织、管理机构，主要负责协调、制定 PKI 相关政策，监督和管理政策的实施，PKI 体系的标准化工作；批准政府根 CA 证书机构和公众服务证书机构（SCA）的设立和证书策略（Certification Policies，CP）与认证操作规范（Certification Practice Statement，CPS），保证其法律效力；负责安全策略的审查及核准。目前，PKI/CA 相关的国家法律法规、地方法律法规、PKI 体系的标准化工作及建立“CA 互联互通示范工程”（这是建设国家级 PKI 安全认证体系的必由之路）正在顺利推进。

2. 政府根 CA

根据国家 PKI 协调管理委员会的相关政策，运营全国电子政务认证体系的根 CA 证书机构，其主要职责包括：运行管理电子政务根 CA 证书机构；批准政府各部委和省（或直辖

市)建立用于政府内部办公的 CA 证书机构和 CP 与 CPS；为各部委、地区的 CA 证书机构签发证书，并查询它们的安全状态。

3. 各行业和地区 CA 证书机构

负责管理用于内部认证的各级 CA 证书机构和 RA。

4. 公众服务证书机构 SCA

各级政府在参与社会活动或与其他公民发生相关业务时，作为组织法人(比如一些公司、行政机关等，与公民处于同等的法律地位)进入相应的 SCA 认证体系，享有与公民对等的义务，承担对等的法律责任。

5. 国家 CA 桥接中心 NBCA

根据国家 PKI 协调管理委员会的相关政策和规范，作为 SCA 认证体系的桥接中心，对由各行业、各区域建立的 SCA 进行交换。代表国家与国外 CA 证书机构进行互认。

6. 不同行业和区域 CA

可以根据国家 PKI 协调管理委员会规定的有关政策和规范，设立相应的 CA 证书机构，如金融、证券、电信、外贸等行业 CA 证书机构和区域性 CA 证书机构。行业 CA 证书机构和区域 CA 证书机构均可以面向全国发放证书，提供相应的信息安全服务。

8.3.3 PKI 的目标

PKI 就是一种基础设施，其目标就是要充分利用公钥密码学的理论基础，建立起一种普遍适用的基础设施，为各种网络应用提供全面的安全服务。公开密钥密码为人们提供了一种非对称性质，使得安全的数字签名和开放的签名验证成为可能。而这种优秀技术的使用却面临着理解困难、实施难度大等问题。正如让电视机的开发者理解和维护发电厂有一定的难度一样，要让每一个应用程序的开发者完全正确地理解和实施基于公开密钥密码的安全有一定的难度。PKI 希望通过一种专业的基础设施的开发，让网络应用系统的开发人员从烦琐的密码技术中解脱出来而同时享有完善的安全服务。

将 PKI 在网络信息空间的地位与电力基础设施在工业生活中的地位进行类比可以更好地理解 PKI。电力基础设施，通过伸到用户的标准插座为用户提供能源，而 PKI 通过延伸到用户本地的接口，为各种应用提供安全的服务。有了 PKI，安全应用程序的开发者可以不用再关心那些复杂的数学运算和模型，而直接按照标准使用一种插座(接口)。正如电冰箱的开发者不用关心发电机的原理和构造一样，只要开发出符合电力基础设施接口标准的应用设备，就可以享受基础设施提供的能源。

PKI 与应用的分离也是 PKI 作为基础设施的重要标志。正如电力基础设施与电器的分离一样。网络应用与安全基础实现了分离，有利于网络应用更快地发展，也有利于安全基础设施更好地建设。正是由于 PKI 与其他应用能够很好地分离，才使得人们能够将之称为基础设施，PKI 也才能从千差万别的安全应用中独立出来，才能有效地独立地发展壮大。PKI 与网络应用的分离实际上就是网络社会的一次“社会分工”，这种分工可能会成为网络

应用发展史上的重要里程碑。

8.3.4 PKI 技术包含的内容

PKI 在公开密钥密码的基础上，主要解决密钥属于谁，即密钥认证的问题。在网络上证明公钥是谁的，就如同在现实中证明谁是什么名字一样具有重要的意义。通过数字证书，PKI 很好地证明了公钥是谁的。PKI 的核心技术就是围绕着数字证书的申请、颁发、使用与撤销等整个生命周期进行展开的。其中，证书撤销是 PKI 中最容易被忽视，但却是很关键的技术之一，也是基础设施必须提供的一项服务。

PKI 技术的研究对象包括数字证书，颁发数字证书的证书认证中心，持有证书的证书持有者和使用证书服务的证书用户，以及为了更好地成为基础设施而必须具备的证书注册机构、证书存储和查询服务器，证书状态查询服务器，证书验证服务器等。

PKI 作为基础设施，两个或多个 PKI 管理域的互联就非常重要。PKI 域间如何互联，如何更好地互联就是建设一个无缝的大范围网络应用的关键。在 PKI 互联过程中，PKI 关键设备之间，PKI 末端用户之间，网络应用与 PKI 系统之间的互操作与接口技术就是 PKI 发展的重要保证，也是 PKI 技术的研究重点。

8.3.5 PKI 的优势

PKI 作为一种安全技术，已经深入到网络的各个层面。这从一个侧面反映了 PKI 强大的生命力和无与伦比的技术优势。PKI 的灵魂来源于公钥密码技术，这种技术使得“知其然，不知其所以然”成为一种可以证明的状态，使得网络上的数字签名有了理论上的安全保障。围绕着如何用好这种非对称密码技术，数字证书破壳而出，并成为 PKI 中最为核心的元素。

PKI 的优势主要表现在：

(1) 采用公开密钥密码技术，能够支持可公开验证并无法仿冒的数字签名，从而在支持可追究的服务上具有不可替代的优势。这种可追究的服务也为原发数据完整性提供了更高级别的担保。支持可以公开地进行验证，或者说任意的第三方可验证，能更好地保护弱势个体，完善平等的网络系统间的信息和操作的可追究性。

(2) 由于密码技术的采用，保护机密性是 PKI 最得天独厚的优点。PKI 不仅能够为相互认识的实体之间提供机密性服务，同时也可以为陌生的用户之间的通信提供保密支持。

(3) 由于数字证书可以由用户独立验证，不需要在线查询，原理上能够保证服务范围的无限制扩张，这使得 PKI 能够成为一种服务巨大用户群的基础设施。PKI 采用数字证书方式进行服务，即通过第三方颁发的数字证书证明末端实体的密钥，而不是在线查询或在线分发。这种密钥管理方式突破了过去安全验证服务必须在线的限制。

(4) PKI 提供了证书的撤销机制，从而使得其应用领域不受具体应用的限制。撤销机制提供了在意外情况下的补救措施，在各种安全环境下都可以让用户更加放心。另外，因为有撤销技术，不论是永远不变的身份、还是经常变换的角色，都可以得到 PKI 的服务而不用担心被窃后身份或角色被永远作废或被他人恶意盗用。为用户提供“改正错误”或“后悔”的途径是良好工程设计中必须的一环。

（5）PKI 具有极强的互联能力。不论是上下级的领导关系，还是平等的第三方信任关系，PKI 都能够按照人类世界的信任方式进行多种形式的互联互通，从而使 PKI 能够很好地服务于符合人类习惯的大型网络信息系统。PKI 中各种互联技术的结合使建设一个复杂的网络信任体系成为可能。PKI 的互联技术为消除网络世界的信任孤岛提供了充足的技术保障。

习题

1．为什么要进行密钥管理？
2．为什么要在密钥管理中引入层次式结构？
3．密钥管理的生命周期包括哪些阶段？
4．密钥的分发方法包括哪些？如何实现？

第9章 操作系统安全技术

9.1 Windows 操作系统安全模型

9.1.1 Windows 系统安全模块

Windows 系统的安全模块是操作系统内核不可分割的一部分。由于访问任何系统资源必须经过内核安全模块的验证，从而保证没有得到正确授权的用户不能访问相应的资源。

用户使用 Windows 系统资源，首先必须在系统中拥有账号，其次，此账号必须具有一定的"权力"和"权限"。在 Windows 系统中，"权力"指用户对整个系统能够做的事情，如关闭系统、增加设备、更改系统时间等。"权限"指用户对系统资源所能做的事情，如对某文件的读、写控制，对打印机队列的管理。Windows 系统使用安全账号数据库，存放用户账号以及该账号所具有的权力等。用户对系统资源所具有的权限则与特定的资源一起存放。

在 Windows 系统中，安全模型由本地安全认证、安全账号管理器和安全监督器构成。除此之外还包括注册、访问控制和对象安全服务等。它们之间的相互作用和集成构成了安全模型的主要部分。

Windows 安全模型的主要功能是用户身份验证和访问控制。身份验证过程通过某种技术手段确认用户所提供的身份的真实性，并在确认用户身份的真实性后赋予用户相应的权利和系统身份标识。访问控制机制利用用户获得的系统身份标识，以及事先分配给用户对系统资源的权限来确保系统资源被合理地使用。

1. 用户身份验证

Windows 安全子系统提供了两种类型的身份验证：通过控制台交互式登录系统（根据用户的计算机的本地账户来确认用户的身份）和通过网络登录系统（根据域控制器中保留的域账户来确认用户的身份）从而使得用户可以访问网络上远程主机的资源。为保证通过网络登录系统的安全性，Windows 安全子系统提供了三种不同的身份验证机制：Kerberos V5、公钥证书和 NTLM。

2. 基于对象的访问控制

Windows 采用对象模型描述系统资源，管理员可以通过对特定资源配置相应的用户访

问权限来控制用户对系统资源的访问。管理员可以通过域控制器实现对整个域的资源的统一管理与控制。Windows 系统允许管理员以对象安全描述符的方式描述具体的访问控制策略。安全描述符列出了允许访问对象的用户和组,以及分配给这些用户和组的特殊权限。安全描述符还指定了该对象需要安全审核特定事件,如特定用户的读,写,执行文件。文件、打印机和服务都是对象的具体例子。通过管理对象的属性,管理员可以设置权限,分配所有权以及监视用户访问。

9.1.2 用户名和密码

Windows 系统的安全机制通过分配用户账号和用户密码来帮助保护计算机及其资源。给值得信任的使用者,按其使用的要求和网络所能给予的服务分配合适的用户账号,并且使用足够安全的账号密码。使用对账号的用户权力的限制以及对文件的访问管理权限的策略,可以达到对服务器数据的保护。其中用户账号有用户名、全名、描述三个部分。用户名是用户账号的标识,全名是对应用户名的全称,描述是对用户所拥有的权限的较具体的说明。组有组名和描述两个部分,组名是标识,描述是说明。一定的用户账号对应一定的权限,NT 对权限的划分比较细,例如备份、远程管理、更改系统时间等,通过对用户的授权(在规则菜单中)可以细化一个用户或组的权限。用户的账号和密码有一定的规则,包括账号长度,密码的有效期,登录失败的锁定,登录的历史记录等,通过对这些的综合修改可以保证用户账号的安全使用。

系统将用户分为管理者、用户和来宾三类,各有其不同的权限。系统在安装完成后自动建立 Administrator(系统管理员)和 Guest(来宾)用户。可以在系统启动后更改系统管理员的密码,还可以单击“添加/删除”来添加/删除用户或用户组。

系统管理员对用户和密码的管理权限主要有添加用户、删除用户及更改用户。系统会在添加新用户时询问其权限的设置,在其中可对此用户账号进行是否允许修改密码、是否停用账号等项的设置。其中,停用账户和删除账户是有区别的,停用账户是临时停止某个账户的使用,随时可以恢复,而删除掉的账户必须重建后才能使用。

另外,Windows 系统支持工作组的概念,可以方便地给一组用户授予特权和权限,同时一个用户同时属于一个或多个工作组。方便了对用户权限的细化。在 Windows 系统中有两种类型的工作组:全局工作组和本地工作组。本地工作组只能在本地的系统或域内使用。全局工作组可以在系统内相互信任的域中使用。

9.1.3 域和委托

以 Windows 系统组建的网络是一个局域网范围的网。所谓“域”是指网络服务器和其他计算机的逻辑分组,凡是在共享域范围内的用户都使用公共的安全机制和用户账号信息。每个用户有一个账号,每次登录的是整个域,而不是某一个服务器。即使在物理上相隔较远,但在逻辑上可以在一个域上,域的集中化用户账号数据库和安全策略使得系统管理员可以用一个简单而有效的方法维护整个网络的安全。在网络环境下,使用域的管理就显得更为有效。这里应该注意到在 NT 中,关于域所用的安全机制信息或用户账号信息都存放在目录数据库中(称为安全账号管理器(SAM)数据库)。目录数据库存放在服务器中,并且复

制到备份服务器中。通过有规律的同步处理，可以保证数据库的安全性、有效性。在用户每次登录时，通过目录数据库检查用户的账号和密码。所以在对 NT 进行维护时应该特别小心目录数据库的完整性，一般来讲只有管理员才具有对此的编辑权限。

域最大的优点是域中的控制器服务器形成了共享的安全机制和用户账号信息的单个管理单元，大大地节省了管理员和用户的精力和时间，在管理上较方便，也显得集中。在使用“域”的划分时，应该注意到“域”是建立在一个子网范围内的，其基础是相互之间的信任度。由 NT 组网区别于一般的 TCP/IP 的组网，TCP/IP 是一种较松散的组网形式，靠路由器完成子网之间的寻径通信；而 NT 组网是一种紧密的联合，服务器之间是靠安全信任建立它们的联系的。主从关系，委托关系是建立在信任度上的。委托是一种管理办法，它将多个域连接在一起，并且允许域中的用户互相访问。委托关系可使用户账号和工作组能够在建立它们的域之外的域中使用。委托关系只能被定义为单向的，为了获得双向委托关系，域和域之间必须相互委托。

9.1.4　存储控制

Windows 系统启动一个用户进程，将存储标识与之连在一起。存取标识包含的内容并没有访问许可权限，而存取标识又是用户在系统中的通行证，那么如何根据存取标识控制用户对资源的访问呢？

当某个进程要访问一个对象时，进程的 SID 与访问控制项列表比较，决定是否可以访问该对象，访问控制列表由访问控制项（ACE）组成，每个访问控制项标识用户和工作组对该对象的访问权限。一般情况下，访问控制列表有三个访问控制项，分别代表以下含义：拒绝对该对象的访问；允许对该对象读取和写入；允许执行该对象。访问控制列表首先列出拒绝访问的访问控制项，然后才是允许的访问控制项。

给资源分配的权限作为该资源的一个属性与资源一起存放。比如目录为 D:\Files，对其指定 User1 只读，User2 可完全控制，则这两个权限都作为 D:\Files 目录的属性与该目录连在一起，在系统内部以访问控制列表的形式存放。包含了每个权限的分配，以访问控制项来表示。

9.2　Windows 操作系统安全设置

这里以 Window 7 操作系统为例进行说明。

9.2.1　检查和删除不必要的账户

单击“开始”按钮，选择“控制面板”中的“用户账户”项；在弹出的对话框中列出了系统的所有账户。确认各账户是否仍在使用，删除其中不用的账户。

例如，图 9-1 中是以 Administrator 管理员登录的界面，单击“管理其他账户”，可以看到如图 9-2 所示的所有用户，可将不用的 a，b 账户删除。

单击 a 用户，可以得到图 9-3 的界面，单击“删除账户”即可。

如果无法删除，可以通过另一种方法删除。

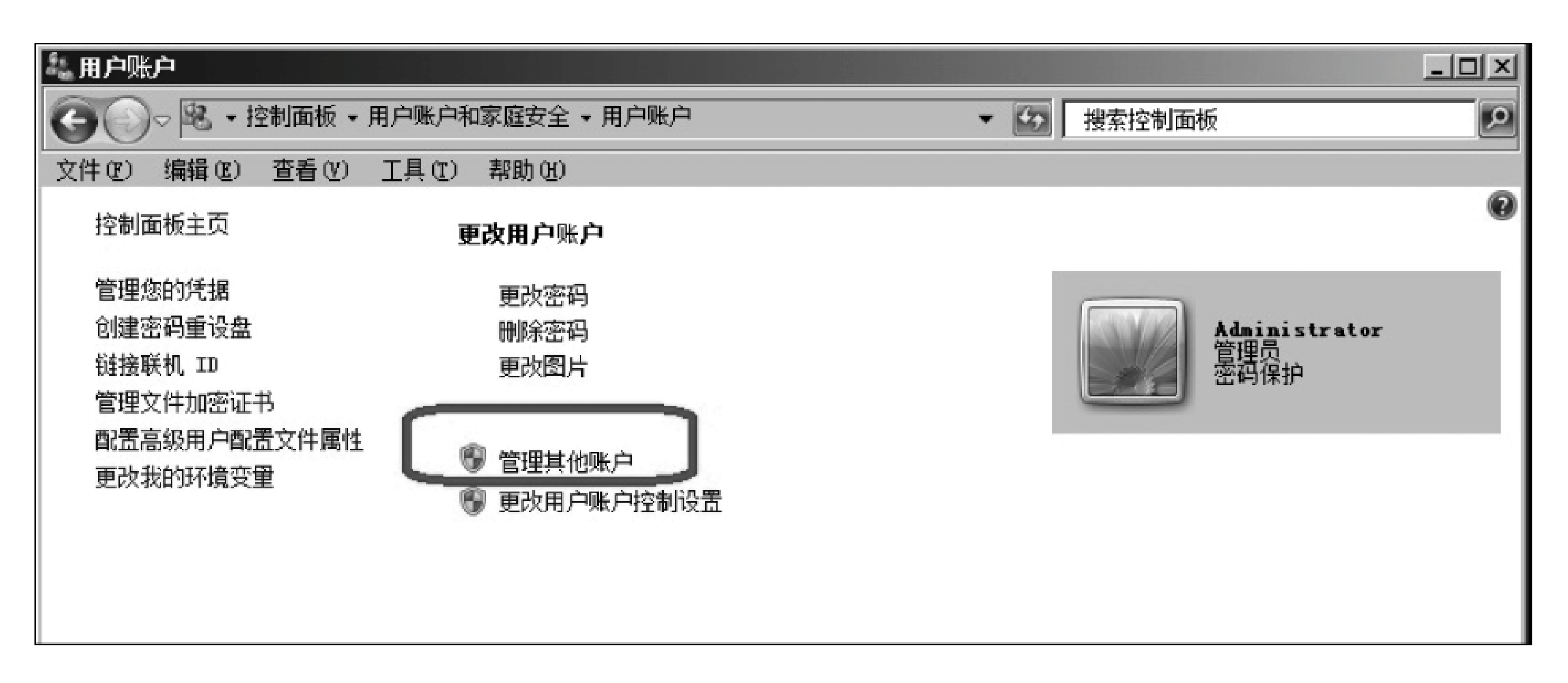


图 9-1 Administrator 管理员登录界面



图 9-2 所有用户

右击“我的电脑”,选择“管理”项,打开计算机管理窗口,打开本地用户和组文件夹,右击要删除的账户名,在弹出的菜单中,选择删除即可,如图 9-4 所示。

9.2.2 停止启用来宾 Guest 账户

在控制面板中的“用户账户”的对话框中,停止启用 Guest 账户。或者在图 9-4 中,右击 Guest 图标,选择“属性”项,在打开的对话框中,选择“账户已禁用”即可,如图 9-5 所示。



图 9-3　用户 a

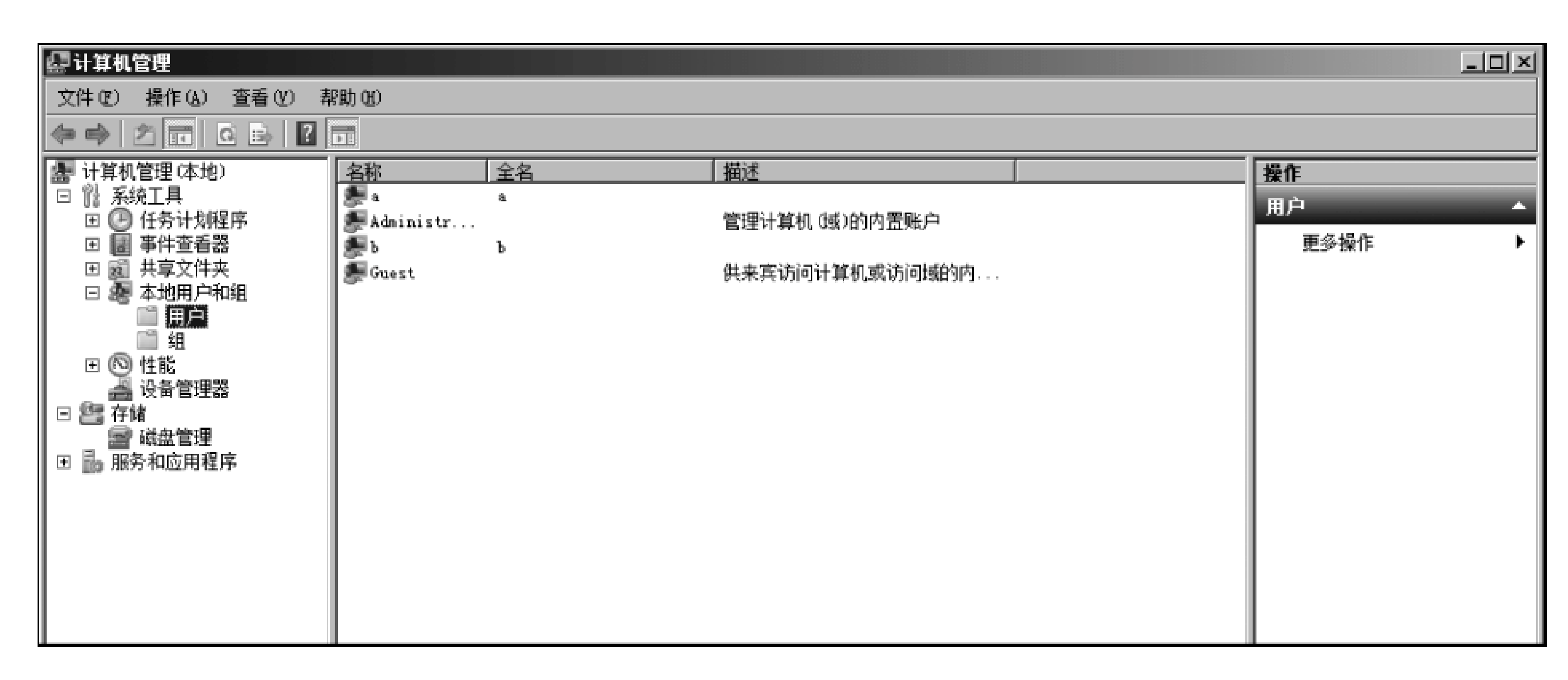


图 9-4　计算机管理-用户



图 9-5　Guest 用户设置

9.2.3 锁定无效登录

为了防止他人进入电脑时，反复用猜测密码的方式登录，可以锁定无效登录，当密码输入错误达设定次数后，便锁定此账户，在一定时间内不能再以该账户登录。

进入控制面板，依次展开"系统与安全" →"管理工具"→"本地安全策略"，出现"本地安全策略"窗口，在左侧列表中打开"账户策略"→"账户锁定策略"，右侧出现如图 9-6 所示的界面。



图 9-6 账户锁定策略

在右边双击"账户锁定阈值"，在弹出的设置对话框中输入无效登录的次数(一般设 3 次为宜)，即如果输入密码错误 3 次，相应账户将自动锁定，具体说明为"此安全设置确定导致用户账户被锁定的登录尝试失败的次数。在管理员重置锁定账户或账户锁定时间期满之前，无法使用该锁定账户。可以将登录尝试失败次数设置为介于 0 和 999 之间的值。如果将值设置为 0，则永远不会锁定账户。"当进行设置时弹出对话框"建议的数值改动"，如图 9-7 所示，确定即可，如果认为建议的锁定时间不满意，可以双击"账户锁定时间"进行更改，如图 9-8 所示。

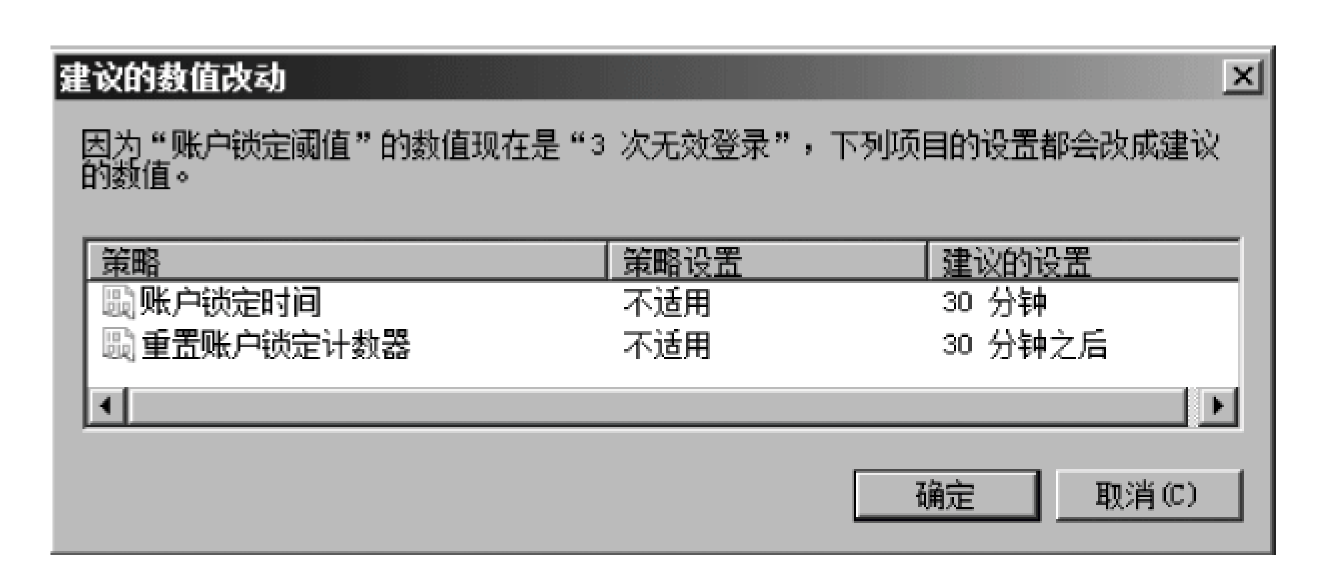


图 9-7 建议的数值改动

经过以上设置，就能够阻止那些靠猜密码登录的非法用户了。

9.2.4 加强密码安全

为了让各账户的密码相对安全、不易被破解，可设置密码策略，加强密码的安全性，最有

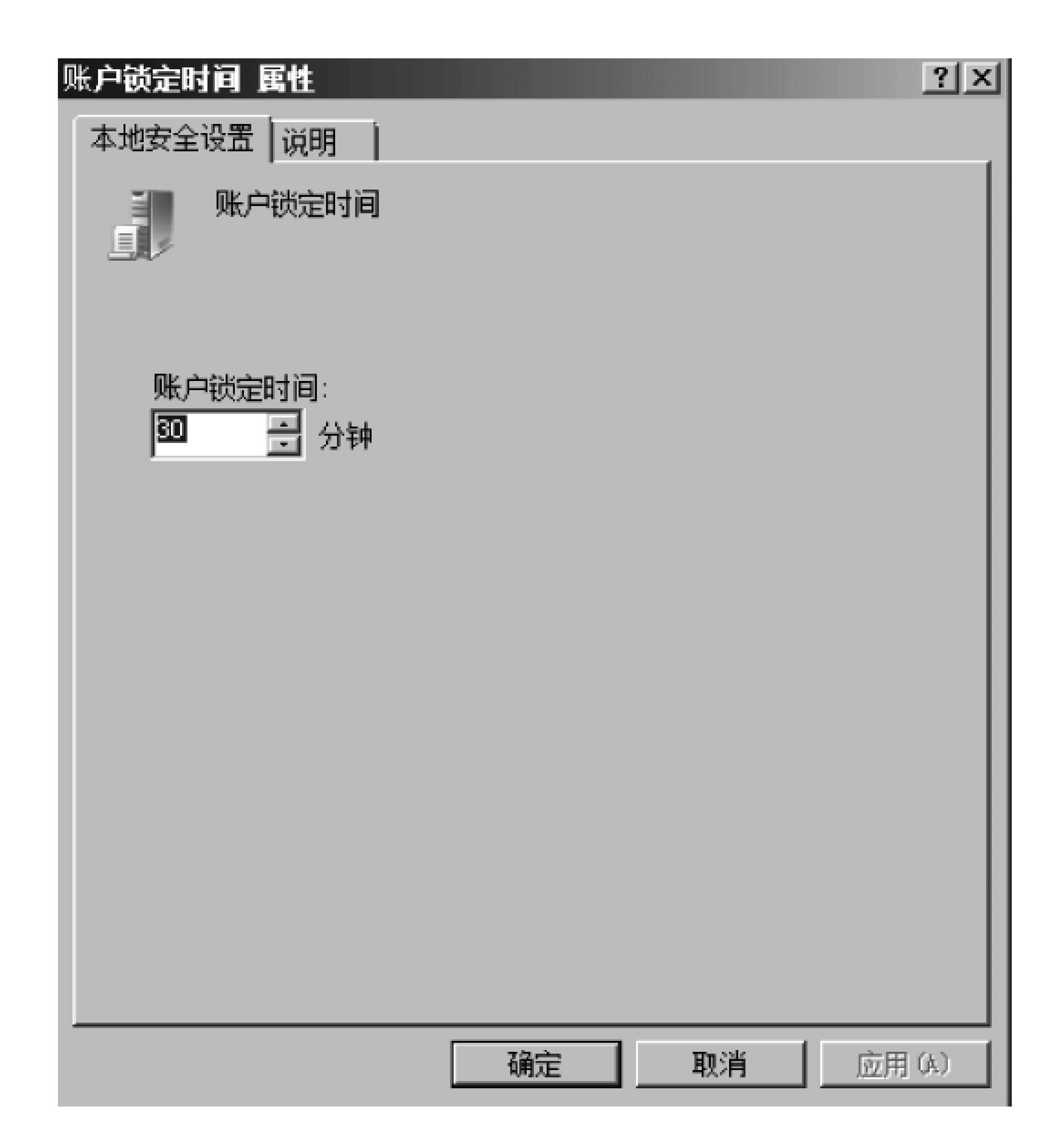


图 9-8 账户锁定时间

效的方法是增加密码的长度和复杂性，并定期更改密码。

进入控制面板，依次展开“系统与安全”→“管理工具”→“本地安全策略”，出现“本地安全策略”窗口，在左侧列表中打开“账户策略”→“密码策略”，如图 9-9 所示。



图 9-9 密码策略

打开“密码长度最小值”，设置最少字符数为 8 位以上。

打开“密码必须符合复杂性要求”，设置为“已启动”。

打开“密码最长使用期限”设置密码能使用的天数。

注意，在窗口中可查看上面设置的内容说明，经过以上三项设置后，凡新建账户或老账户更改密码时，系统会要求使用 8 位以上同时包含英文字母、数字或标点的密码，并且必须按设定的时间定期更改密码，密码的安全性将大大增强。

说明：如果感觉密码输入太复杂而不方便时，可不设置以上的内容。

9.2.5 设置账户名保密

默认情况下，在系统登录框中会保留上次登录的用户名，这方便了该用户的登录，但却留下了安全隐患，特别是对管理员账户，暴露账户名称是一件非常危险的事情，因为有不良企图者只需输入密码便可尝试登录，甚至使用专门的工具来攻破此账户。可以将上次登录账户隐藏起来。

进入控制面板，依次展开“系统与安全”→“管理工具”→“本地安全策略”，出现“本地安全策略”窗口，单击“安全选项”，如图 9-10 所示。在右边找到“不显示最后的用户名”，双击打开此策略，将其设置为“已启用”。



图 9-10 安全选项

进行此项设置后，系统启动或注销后，登录框中用户名为空，必须输入完整有效的用户名和密码才能登录。

9.2.6 更改 Administrator 账户的名字

将计算机默认的管理员名修改，以达到安全效果。

右击“我的电脑”，选择“管理”→“本地用户和组”→“用户”，右击 Administrator，如图 9-11 所示，重命名输入所需要的名字即可。

9.2.7 禁止枚举账号

由于某些具有黑客行为的蠕虫病毒可以通过扫描系统的指定端口，然后共享会话猜测管

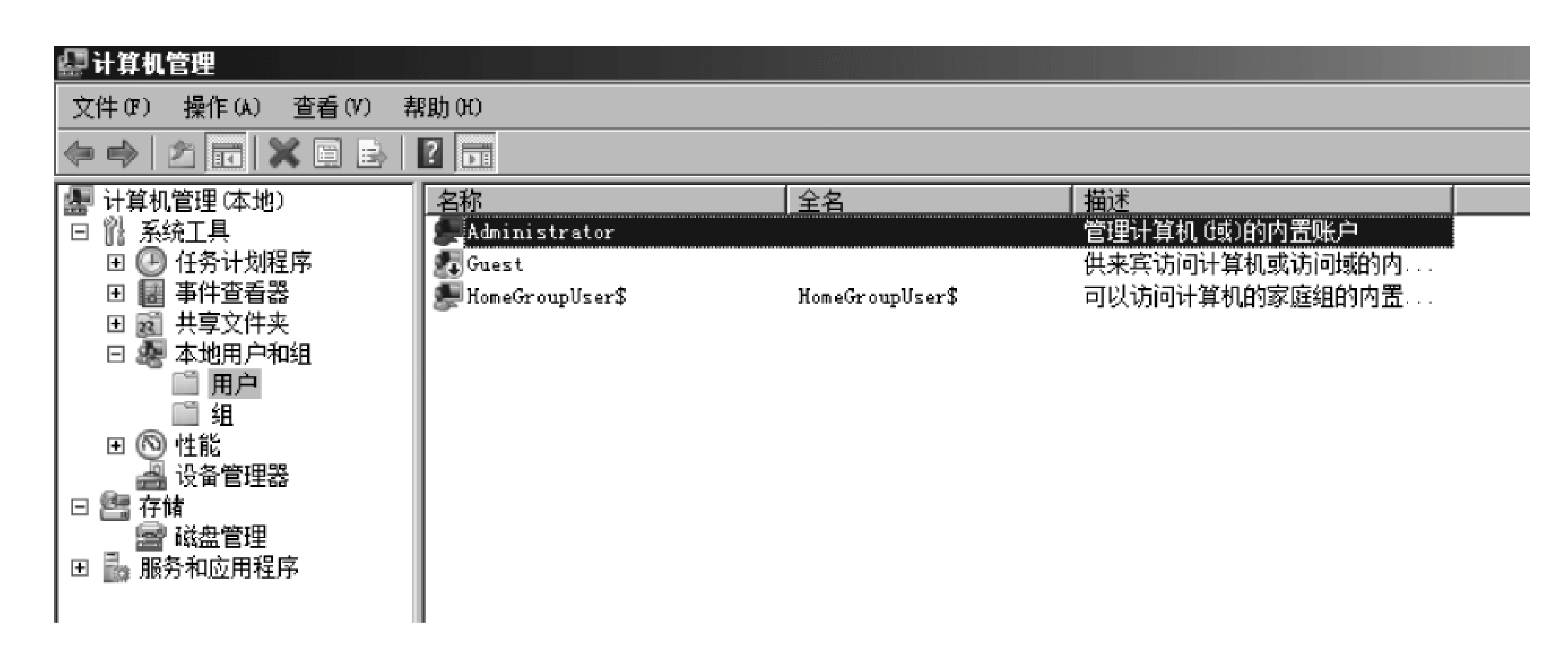


图 9-11　更改 Administrator 账户的名字

理系统口令，所以要用户设置本地安全策略，来禁止枚举账号，从而能抵御此类入侵行为。

进入控制面板，依次展开“系统与安全”→“管理工具”→“本地安全策略”，出现“本地安全策略”窗口，单击“安全选项”，在右边找到“不允许 SAM 账户匿名枚举”，双击打开此策略，将其设置为“已启用”，同时还要对后面的“不允许 SAM 账户和共享的匿名枚举”选择启用，如图 9-12 所示。



图 9-12　安全选项

9.2.8　停止 Schedule 服务

Windows 的 Schedule 服务可以帮助系统管理员设计一个在某个时间执行的批任务。由于 Schedule 服务通常在系统账号下执行，它可以修改账号的权限。这就意味着入侵者可以修改 Schedule 配置并放入一个 TROJAN 木马程序来修改网络的访问权限。

进入控制面板，依次展开“系统与安全”→“管理工具”→“任务计划程序”，出现图 9-13 的窗口。在右侧可以看到很多选项，包括“创建新的、查看已有的”，可以删除所有的计划任务。停止 Schedule 服务，不会对系统造成任何不良的影响。



图 9-13 计划任务

9.2.9 登录前显示一条警示信息

利用此项功能可以在登录前提示一些警示信息或注意事项，以保持系统的正常安全运行。同时防止用户对远程终端服务口令进行自动化的脚本猜测。

进入控制面板，依次展开"系统与安全"→"管理工具"→"本地安全策略"，出现"本地安全策略"窗口，单击"安全选项"，在右边找到"试图登录的用户的消息文本"，如图 9-14 所示。双击打开此策略，出现图 9-15 的对话框，输入警示话语即可。



图 9-14 安全选项

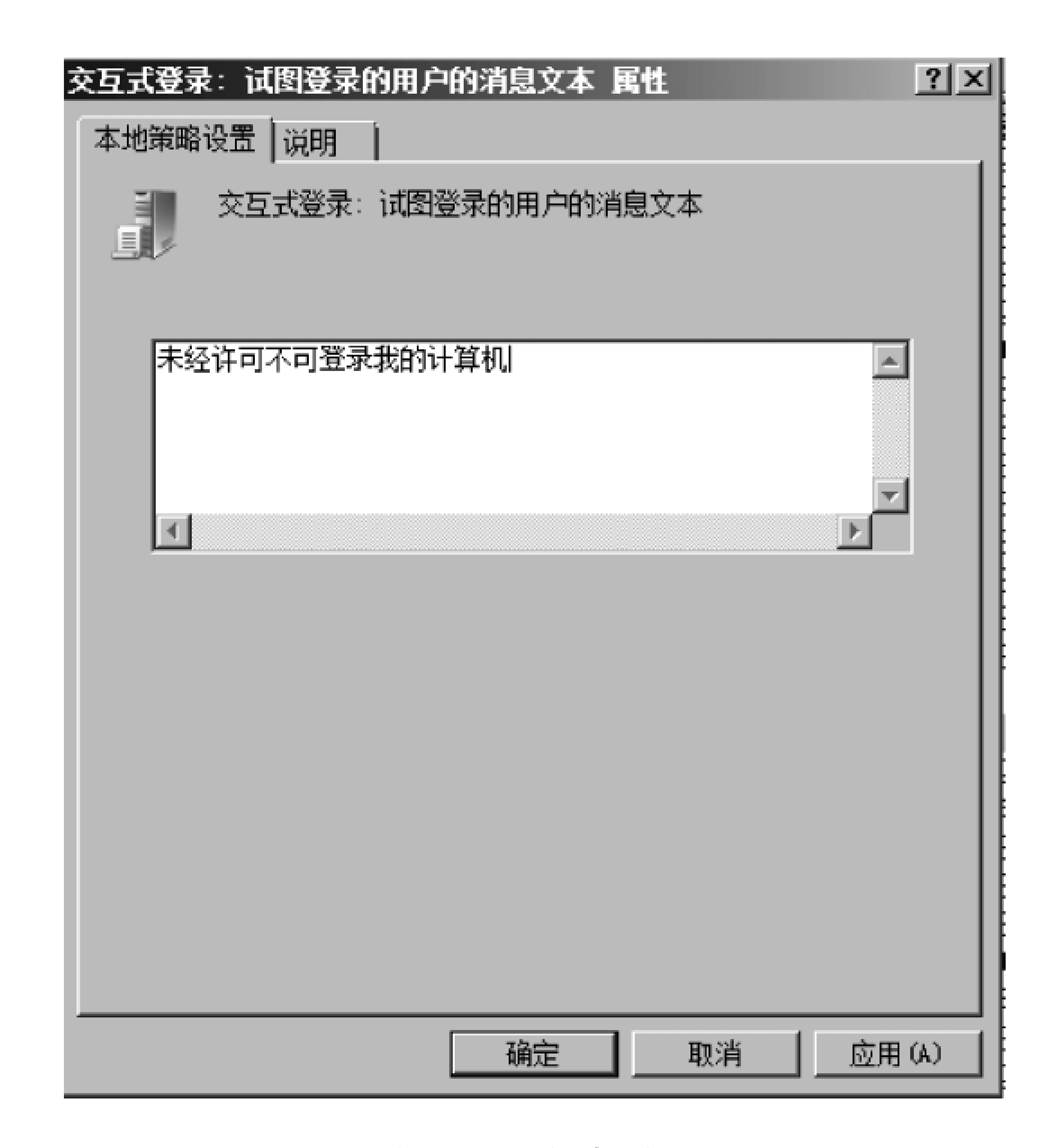


图 9-15　文本设置

9.2.10　从登录对话框中删除关机按钮

如果在登录界面上出现“关机”按钮的话，所有能够接触到该主机的用户都可以关闭机器，这是极其危险的，因此建议在登录界面上删除“关机”按钮。

进入控制面板，依次展开“系统与安全”→“管理工具”→“本地安全策略”，出现“本地安全策略”窗口，单击“安全选项”，在右边找到“允许系统在未登录的情况下关闭”，如图 9-16 所示。双击打开此策略，改成“已禁用”。

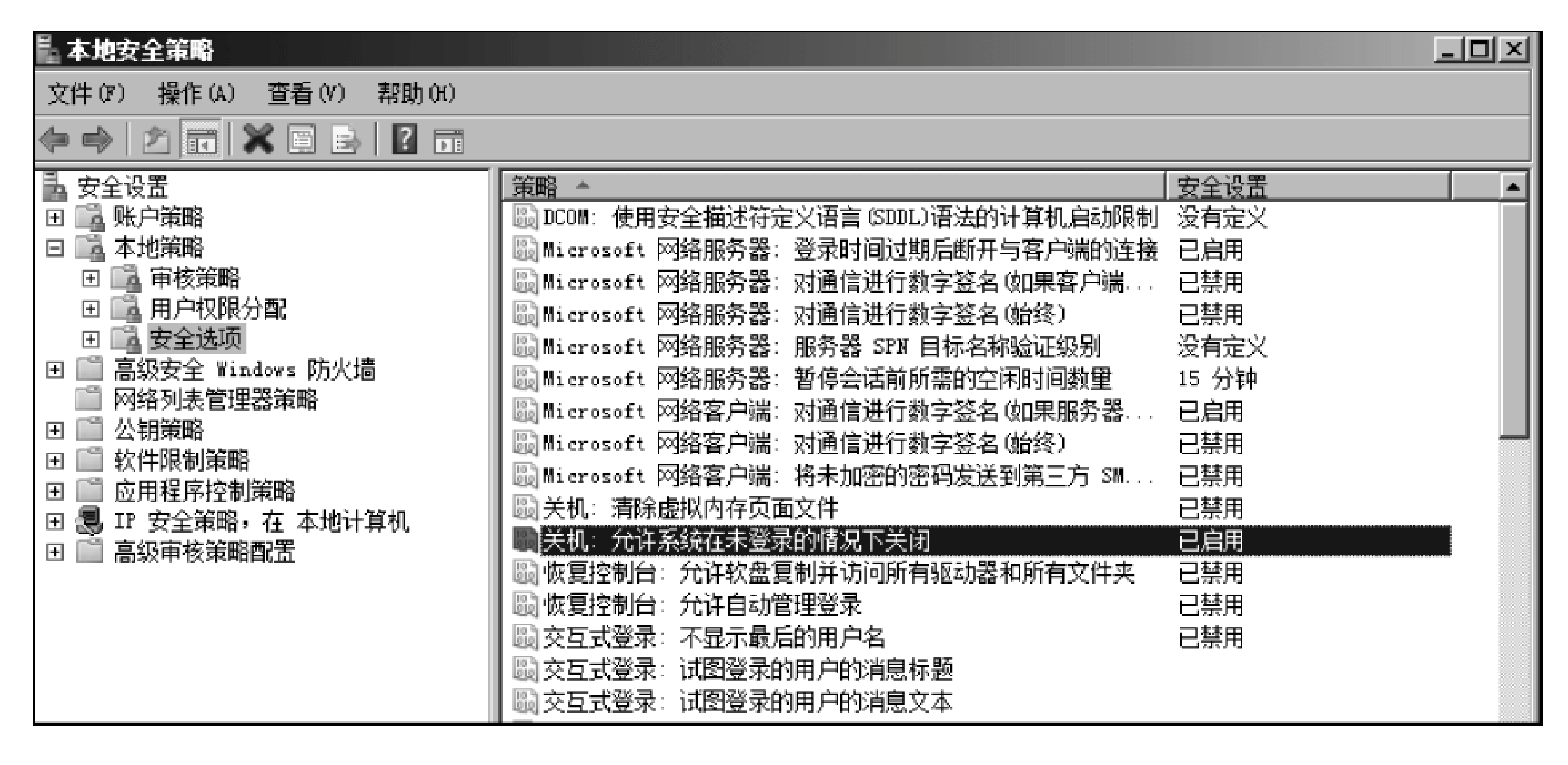


图 9-16　安全选项



习题

1. 叙述计算机系统安全的主要内容。
2. 安全操作系统的主要功能有哪些？
3. 计算机或网络系统在安全性上受到的威胁有哪些？

第10章 计算机病毒与木马

10.1 计算机病毒概述

一般来讲，凡是能够引起计算机故障，能够破坏计算机中的资源(包括硬件和软件)的代码，统称为计算机病毒。美国国家计算机安全局出版的《计算机安全术语汇编》对计算机病毒的定义是："计算机病毒是一种自我繁殖的特洛伊木马，它由任务部分、接触部分和自我繁殖部分组成"。而在我国也通过条例的形式给计算机病毒下了一个具有法律性、权威性的定义，《中华人民共和国计算机信息系统安全保护条例》明确定义："计算机病毒(Computer Virus)是指编制或者在计算机程序中插入的破坏计算机功能或者数据，影响计算机使用并且能够自我复制的一组计算机指令或者程序代码"。

10.1.1 计算机病毒的特性

1. 计算机病毒的非授权可执行性

计算机病毒与其他合法程序一样，是一段可执行程序，但它不是一个完整的程序，而是寄生在其他可执行程序上的，因为它享有一切程序所能得到的权力。在病毒运行时，与合法程序争夺系统的控制权。计算机病毒只有当它在计算机内得以运行时，才具有传染性和破坏性。也就是说计算机 CPU 的控制权是关键问题。

用户通常调用执行一个程序时，把系统控制交给这个程序，并分配给它相应的系统资源，如内存，从而使之能够运行完成用户的需求。因此程序执行的过程对用户是透明的。而计算机病毒是非法程序，正常用户是不会明知是病毒程序，而故意调用执行的。但由于计算机病毒具有正常程序的一切特性，即可存储性、可执行性。它隐藏在合法的程序或数据中，当用户运行正常程序时，病毒伺机窃取到系统的控制权，得以抢先运行，然而此时用户还认为在执行正常程序。

2. 计算机病毒的隐蔽性

计算机病毒是一种具有很高编程技巧、短小精悍的可执行程序代码或者一个独立存在的程序，为了防止被发现，总是千方百计地将自己隐藏起来。如不经过程序代码分析或计算机病毒代码扫描，病毒程序与正常程序是不容易区别开来的。

计算机病毒的隐蔽性表现为两个方面：

(1) 传染的隐蔽性。一般不具有外部表现,不易被人发现。

(2) 病毒程序存在的隐蔽性。一般的病毒程序都夹在正常程序之中,很难被发现,而一旦病毒发作出来,往往已经给计算机系统造成了不同程度的破坏。

从计算机病毒隐藏的位置来看,不同的病毒隐藏在不同的位置。位于系统引导区的病毒通常不需要隐藏,因为引导区内的代码通常是不可见的;依附在文件上的病毒由于有文件做掩护,通常也不需要特殊的隐藏措施,但是文件被病毒附身后尺寸往往会增大,一些隐蔽性好的病毒会将自身藏匿在这些文件内部未被使用的空隙中,使感染后的文件尺寸并未增大,隐蔽性得到加强;独立存在的病毒没有宿主程序做掩护,隐蔽性很差,通常仅采取将病毒程序加上隐藏和系统属性,在 Windows 系统中进行合理的配置就可以使这些病毒现身。

3. 计算机病毒的传染性

传染是病毒最重要的特征,是判断一段程序代码是否为计算机病毒的依据。计算机病毒依靠其传染性不断将自己复制和扩散,这是计算机病毒和特洛伊木马程序的显著区别。

计算机病毒是一段人为编制的计算机程序代码,这段程序代码一旦进入计算机并得以执行,它就会搜索其他符合其传染条件的程序或者存储介质,确定目标后再将自身代码插入其中,达到自我繁殖的目的。只要一台计算机感染病毒,如不及时处理,那么病毒会在这台计算机上迅速扩散,其中的大量文件会被感染。而被感染的文件又成了新的传染源,再与其他机器进行数据交换或通过网络接触,病毒会继续进行传染。在某些情况下造成被感染的计算机工作失常甚至瘫痪,因此,这也是计算机病毒这一名称的由来。

计算机病毒的传染方式有多种,通常采取通过接管系统中断的方法传染。所有的传染都是在病毒程序被启动后才能够进行的。未被启动的病毒程序是不会传染的。病毒程序被启动后抢先接管系统中断,当有磁盘或文件读写等操作时首先被发现,病毒程序判断是否需要对这个磁盘或文件进行感染,如果需要则病毒程序自动将病毒代码写入此磁盘或文件中,完成病毒的感染过程,传染病毒完成后病毒程序再对磁盘或文件进行正常的读写操作,使用户无法发现病毒的传染。有些病毒接管网络通信中断,当进行网络通信时这些病毒将病毒代码通过网络通信传输到远方目标计算机,完成病毒的传染过程。

4. 计算机病毒的潜伏性

计算机病毒为了更广泛地传播和扩散,通常完成传染过程后不会立即发作进行破坏活动,而是将自己深深地隐藏起来,伺机进行传染。一个计算机病毒的潜伏性越好,其在系统中存在的时间就会越长,病毒的传染范围就会越广,其危险性和破坏性就越大。

计算机病毒潜伏性的第一种表现是指,病毒程序不用专用检测程序是检查不出来的,因此病毒可以潜伏在磁盘或磁带里几天,甚至几年,一旦时机成熟,得到运行机会,就又要四处繁殖、扩散,继续危害;第二种表现是指,计算机病毒的内部往往有一种触发机制,不满足触发条件时,计算机病毒除了传染外没有别的破坏,触发条件一旦得到满足,有的在屏幕上显示信息、图形或特殊标识,有的则执行破坏系统的操作,如格式化磁盘、删除磁盘文件、对数据文件做加密、封锁键盘以及使系统死锁等。

5. 计算机病毒的破坏性

按计算机病毒的破坏性可将其分为主动进行破坏的恶性病毒和不主动进行破坏的良性病毒。

恶性病毒的破坏性很明显，如删除磁盘文件、破坏磁盘分区、破坏CMOS内容甚至破坏BIOS内引导程序。这些破坏会给用户造成巨大的损失，会使用户多年的心血毁于一旦，这一类病毒令广大计算机用户深恶痛绝。还有一类计算机病毒并不主动进行破坏，不破坏用户的文件资源，也不会造成系统死机和崩溃，甚至有时还会给用户播放一段优美的乐曲或不定时地弹出一个问候语。这些计算机病毒虽然不直接对系统进行破坏，但是其运行时必定要占用用户的系统资源，降低用户计算机系统的工作效率，而这种占用是未经用户允许的。所有未经用户允许就擅自侵占用户资源的行为都是对用户系统资源的侵占和破坏。因此说所有的计算机病毒都具有破坏性。

计算机病毒的破坏性取决于计算机病毒制造者的目的和水平，它可以直接破坏计算机数据信息、抢占系统资源、影响计算机运行速度以及对计算机硬件构成破坏等。病毒程序的表现性或破坏性体现了病毒设计者的真正意图。正是由于计算机病毒可怕的破坏性才使得计算机病毒如此恐怖。

6. 计算机病毒的寄生性

计算机病毒通常不是以一个独立的程序出现在计算机系统中的，而是附着在计算机的操作系统、各个可执行文件等的宿主程序中生存，因而在其潜伏阶段不易被人察觉。

7. 计算机病毒的衍生性

计算机病毒的衍生性是指计算机病毒的制造者依据个人的主观愿望，对某一个已知病毒程序进行修改而衍生出另外一种或多种来源于同一种病毒，而又不同于源病毒程序的病毒程序，即源病毒程序的变种。病毒的衍生性为一些好事者提供了一种创造新病毒的捷径。这也许就是病毒种类繁多、复杂的原因之一。

8. 计算机病毒的可触发性

计算机病毒因某个事件或数值的出现，诱使病毒实施感染或进行攻击的特性称为可触发性。

计算机病毒为了隐藏自己，必须潜伏，少做动作。但如果完全不动，一直潜伏，病毒就既不能感染也不能进行破坏，便失去了杀伤力。

病毒既要隐藏又要维持杀伤力，必须具有可触发性。病毒的触发机制就是用来控制感染和破坏动作的频率的。病毒具有预定的触发条件，这些条件可能是时间、日期、文件类型或某些特定数据等。病毒运行时，触发机制检查预定条件是否满足，如果满足，启动感染或破坏动作，使病毒进行感染或攻击；如果不满足，使病毒继续潜伏。

10.1.2 计算机病毒的传播途径

传染性是计算机病毒最基本的特性，也是病毒赖以生存繁殖的条件，如果计算机病毒没

有传播渠道，则其破坏性小，扩散面窄，难以造成大面积流行。因此计算机病毒必须要“搭载”到计算机上才能感染系统，通常它们是附加在某个文件上的。

处于潜伏期的病毒在激发之前，不会对计算机内的信息进行破坏，即绝大部分磁盘信息没有遭到破坏。因此，只要消除没有发作的计算机病毒，就可保护计算机的信息。病毒的复制与传染过程只能发生在病毒程序代码被执行过后。也就是说，虽然有一个带有病毒程序的文件存储在计算机硬盘上，但是只要永远不去执行它，这个计算机病毒也就永远不会感染计算机。从用户的角度来说，只要能保证所执行的程序是“干净”的，那计算机就绝不会染上病毒，但是由于计算机系统自身的复杂性，许多用户是在不清楚所执行的程序是否可靠的情况下执行程序的，这就使得病毒入侵的机会大大增加，并得以传播扩散。

计算机病毒的传播主要通过文件复制、文件传送、文件执行等方式进行，文件复制与文件传送需要传输媒介，文件执行则是病毒感染的必然途径（Word、Excel 等宏病毒通过 Word、Excel 调用间接地执行），因此，病毒传播与文件传输媒体的变化有着直接关系。通过认真研究各种计算机病毒的传染途径，有的放矢地采取有效措施，必定能在对抗计算机病毒的斗争中占据有利地位，更好地防止病毒对计算机系统的侵袭。计算机病毒的主要传播途径如下。

1. 光盘

光盘因为容量大，可以存储大量的可执行文件，而大量的病毒就有可能藏身于光盘中。对于只读式光盘，由于不能进行写操作，因此光盘上的病毒不能清除。在以谋利为目的非法盗版软件制作过程中，不可能为病毒防护担负专门责任，也决不会有真正可靠的技术保障避免病毒的侵入、传染、流行和扩散。当前，盗版光盘的泛滥给病毒的传播带来了极大的便利，甚至有些光盘上的杀病毒软件本身就带有病毒，这就给本来“干净”的计算机带来了灾难。

2. 硬盘

有时，带病毒的硬盘（含移动硬盘、U 盘）会在本地或移到其他地方使用甚至维修等，这就会传染干净的软盘或者感染其他硬盘并扩散病毒。

3. 有线网络

现代通信技术的巨大进步已使空间距离不再遥远，数据、文件、电子邮件可以方便地在各个网络工作站间通过电缆、光纤或电话线路进行传送，工作站的距离可以短至并排摆放的计算机，也可以长达上万千米，这在为人们带来便利的同时也为计算机病毒的传播提供了新的载体。计算机病毒可以附着在正常文件中，当从网络另一端得到一个被感染的程序，并在计算机上未加任何防护措施的情况下运行它时，病毒就传染开来。这种病毒传染方式常见于在计算机网络连接很普及的国家，国内计算机感染一些“进口”病毒已不再是什么大惊小怪的事了。在信息国际化的同时，病毒也在国际化。大量的国外病毒随着因特网传入国内。

网络的快速发展促进了以网络为媒介的各种服务（FTP，WWW，BBS，E-mail 等）的快速普及。同时，这些服务也成了新的病毒传播方式。

电子布告栏(BBS)：BBS是由计算机爱好者自发组织的通信站点，用户可以在BBS上进行文件交换(包括自由软件、游戏、自编程序)。由于大多数BBS网站没有严格的安全管理，亦无任何限制，这样就给一些病毒程序编写者提供了传播病毒的场所。各城市BBS站点通过中心站点进行传送，传播面较广。BBS在国内的普及，给病毒的传播又增加了新的介质。

电子邮件(E-mail)：计算机病毒主要以附件的形式进行传播，由于人们可以发送任何类型的文件，而大部分计算机病毒防护软件在这方面的功能还不是十分完善，使得电子邮件成为当今世界上传播计算机病毒最主要的媒介。

即时消息服务(QQ，ICQ，MSN等)：像电子邮件一样，消息服务同样可以自由地传播文件，从而也成为计算机病毒传播的主要途径之一。

Web服务：Web网站在传播有益信息的同时，也成了传播不良信息的重要途径。Script和ActiveX技术被广泛用来编制病毒和恶意攻击程序，它们主要通过Web网站传播，不法分子或好事之徒制作的匿名个人网页直接提供了下载大批病毒活样本的便利途径。散见于网站上大批的病毒制作工具、向导、程序等，使得无编程经验者制造新病毒成为可能。新技术、新病毒使得几乎所有人在不知情时无意中成为病毒扩散的载体或传播者。

FTP服务：通过这个服务，可以将文件放在世界的任何一台计算机上，或者从计算机复制到本地机器上。这很大程度上方便了学习和交流，成为因特网上别有用心的人使用的工具。关于病毒制作研究讨论的学术性质的电子论文、期刊、杂质得到最大程度的共享，但同时也使因特网上的病毒传播更容易、更广泛。这一途径能传播现有的所有病毒，所以在使用FTP时就更要注意防毒。

新闻组：通过这种服务，可以与世界上的任何人讨论某个话题，或选择接收感兴趣的有关新闻邮件。但这些信息当中包含的附件有可能使计算机感染病毒。

4. 无线通信系统

无线网络已经越来越普及，但很少有无线装置拥有防毒功能。由于未来有更多的手机通过无线通信系统和因特网连接，因此手机已成为电脑病毒的下一个攻击目标。病毒一旦发作，手机就会出现故障。

病毒对手机的攻击有三个层次：攻击WAP服务器，使手机无法访问服务器；攻击网关，向手机用户发送大量垃圾信息；直接对手机本身进行攻击，有针对性地对其操作系统和运行程序进行攻击，使手机无法提供服务。

上面讨论了计算机病毒的传染渠道，随着各种反病毒技术的发展和人们对病毒各种特性的了解，通过对各条传播途径的严格控制，来自病毒的侵扰会越来越少。

10.1.3 计算机病毒的分类

计算机病毒技术的发展，病毒特征的不断变化，给计算机病毒的分类带来了一定的困难。根据多年来对计算机病毒的研究，按照不同的体系可对计算机病毒进行以下分类。

1. 按病毒存在的媒体

根据病毒存在的媒体，病毒可以划分为网络病毒、文件病毒、引导型病毒和混合型病毒。

(1) 网络病毒：通过计算机网络传播感染网络中的可执行文件。

(2) 文件病毒：感染计算机中的文件(如 COM，EXE，DOC 等)。

(3) 引导型病毒：感染启动扇区(Boot)和硬盘的系统引导扇区(MBR)。

(4) 混合型病毒：是上述三种情况的混合。例如，多型病毒(文件和引导型)感染文件和引导扇区两种目标，这样的病毒通常都具有复杂的算法，它们使用非常规的办法侵入系统，同时使用了加密和变形算法。

2. 按病毒传染的方法

根据病毒的传染方法，可将计算机病毒分为引导扇区传染病毒、执行文件传染病毒和网络传染病毒。

(1) 引导扇区传染病毒：主要使用病毒的全部或部分代码取代正常的引导记录，而将正常的引导记录隐藏在其他地方。

(2) 执行文件传染病毒：寄生在可执行程序中，一旦程序执行，病毒就被激活，进行预定活动。

(3) 网络传染病毒：这类病毒是当前病毒的主流，特点是通过因特网络进行传播。例如，蠕虫病毒就是通过主机的漏洞在网上传播的。

3. 按病毒破坏的能力

根据病毒破坏的能力，计算机病毒可划分为无害型病毒、无危险病毒、危险型病毒和非常危险型病毒。

(1) 无害型：除了传染时减少磁盘的可用空间外，对系统没有其他影响。

(2) 无危险型：仅仅是减少内存、显示图像、发出声音。

(3) 危险型：在计算机系统操作中造成严重的错误。

(4) 非常危险型：删除程序、破坏数据、清除系统内存和操作系统中重要的信息。

有些病毒对系统造成的危害，并不是本身的算法中存在危险的调用，而是当它们传染时会引起无法预料的灾难性的破坏。由病毒引起其他的程序产生的错误也会破坏文件和扇区，这些病毒也按照它们引起的破坏能力进行划分。目前的一些无害型病毒也可能会对新版的 DOS、Windows 和其他操作系统造成破坏。例如，在早期的病毒中，有一个名为 Denzuk 的病毒在 360KB 的磁盘上不会造成任何破坏，但是在后来的高密度软盘上却能导致大量的数据丢失。

4. 按病毒算法

根据病毒特有的算法，病毒可以分为伴随型病毒、蠕虫型病毒、寄生型病毒、练习型病毒、诡秘型病毒和幽灵病毒。

(1) 伴随型病毒：这一类病毒并不改变文件本身，它们根据算法产生 EXE 文件的伴随体，具有同样的名字和不同的扩展名(COM)，例如，XCOPY. EXE 的伴随体是 XCOPY.

COM。病毒把自身写入 COM 文件并不改变 EXE 文件,当 DOS 加载文件时,伴随体优先被执行,再由伴随体加载执行原来的 EXE 文件。

(2) 蠕虫型病毒：通过计算机网络传播,不改变文件和资料信息,利用网络从一台机器的内存传播到其他机器的内存,计算网络地址,将自身的病毒通过网络发送。有时它们在系统中存在,一般除了内存不占用其他资源。

(3) 寄生型病毒：依附在系统的引导扇区或文件中,通过系统的功能进行传播。

(4) 练习型病毒：病毒自身包含错误,不能进行很好的传播,例如一些在调试阶段的病毒。

(5) 诡秘型病毒：它们一般不直接修改 DOS 中断和扇区数据,而是通过设备技术和文件缓冲区等对 DOS 内部进行修改,不易看到资源,使用比较高级的技术。利用 DOS 空闲的数据区进行工作。

(6) 幽灵病毒：这一类病毒使用一个复杂的算法,使自己每传播一次都具有不同的内容和长度。它们一般由一段混有无关指令的解码算法和经过变化的病毒体组成。

5. 按病毒的攻击目标

根据病毒的攻击目标,计算机病毒可以分为 DOS 病毒、Windows 病毒和其他系统病毒。

(1) DOS 病毒：指针对 DOS 操作系统开发的病毒。目前几乎没有新制作的 DOS 病毒,由于 Windows 9x 病毒的出现,DOS 病毒几乎绝迹。但 DOS 病毒在 Windows 9x 环境中仍可以进行感染活动,因此若执行染毒文件,Windows 9x 用户的系统也会被感染。人们使用的杀毒软件能够查杀的病毒中一半以上都是 DOS 病毒,可见 DOS 时代 DOS 病毒的泛滥程度。但这些众多的病毒中除了少数几个让用户胆战心惊的病毒之外,大部分病毒都只是制作者出于好奇或对公开代码进行一定变形而制作的病毒。

(2) Windows 病毒：主要指针对 Windows 9x 操作系统的病毒。现在的电脑用户一般都安装 Windows 系统,Windows 病毒一般感染 Windows 9x 系统,其中最典型的病毒有 CIH 病毒。但这并不意味着可以忽略系统是 Windows NT 系列(包括 Windows 2000)的计算机。一些 Windows 病毒不仅在 Windows 9x 上正常感染,还可以感染 Windows NT 上的其他文件。

(3) 其他系统病毒：主要攻击 Linux、UNIX 和 OS2 及嵌入式系统的病毒。由于系统本身的复杂性,这类病毒数量不是很多。

6. 按计算机病毒的链接方式

由于计算机病毒本身必须有一个攻击对象才能实现对计算机系统的攻击,并且计算机病毒所攻击的对象是计算机系统可执行的部分。因此,根据链接方式计算机病毒可分为源码型病毒、嵌入型病毒、外壳型病毒、操作系统型病毒。

(1) 源码型病毒：该病毒攻击高级语言编写的程序,在高级语言所编写的程序编译前插入源程序中,经编译成为合法程序的一部分。

(2) 嵌入型病毒：这种病毒是将自身嵌入现有程序中,把计算机病毒的主体程序与其攻击的对象以插入的方式链接。这种计算机病毒是难以编写的,一旦侵入程序体也较难消

除。如果同时采用多态性病毒技术、超级病毒技术和隐蔽性病毒技术，将给当前的反病毒技术带来严峻的挑战。

(3) 外壳型病毒：外壳型病毒将其自身包围在主程序四周，对原来的程序不做修改。这种病毒最为常见，易于编写，也易于发现，一般测试文件的大小即可察觉。

(4) 操作系统型病毒：这种病毒用自身的程序加入或取代部分操作系统进行工作，具有很强的破坏力，可以导致整个系统的瘫痪。圆点病毒和大麻病毒就是典型的操作系统型病毒。

这种病毒在运行时，用自己的逻辑部分取代操作系统的合法程序模块，根据病毒自身的特点和被替代的合法程序模块在操作系统中运行的地位与作用，以及病毒取代操作系统的取代方式等，对操作系统进行破坏。

10.2 计算机病毒的原理和防范

计算机病毒的基本程序结构如图 10-1 所示。

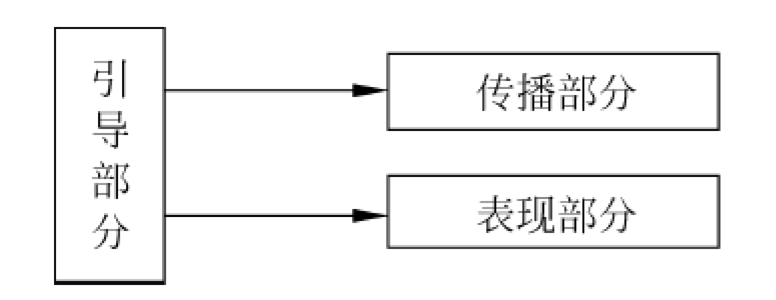


图 10-1 计算机病毒的基本程序结构

(1) 引导部分：把病毒程序加载到内存。

功能：驻留内存、修改中断、修改高端内存、保存原中断向量。

(2) 传染部分：把病毒代码复制到传染目标上。

功能：条件判断、与主程序连接、设置标志。

(3) 表现部分：运行、实施破坏。

功能：条件判断、显示、文件读写。

下面从引导区计算机病毒、文件型计算机病毒、脚本型计算机病毒、特洛伊木马计算机病毒，蠕虫计算机病毒进行分析。

1. 引导区计算机病毒

系统引导区是在系统引导的时候，进入系统中，获得对系统的控制权，在完成其自身的安装后才去引导系统的。称其为引导区计算机病毒是因为这类计算机病毒一般都是侵占系统硬盘的主引导扇区 I/O 分区的引导扇区，对于软盘则侵占了软盘的引导扇区。

它会感染在该系统中进行读写操作的所有软盘，然后再由这些软盘以复制的方式和引导进入其他计算机系统，感染其他计算机的操作系统。

检测的方法包括：

(1) 查看系统内存的总量与正常情况进行比较。

(2) 检查系统内存高端的内容。

(3) 检查系统的 INT 13H 中断向量。

(4) 检查硬盘的主引导扇区、DOS分区引导扇区以及软盘的引导扇区。

清除的方法为：

用原来正常的分区表信息或引导扇区信息，覆盖掉计算机病毒程序。此时，如果用户事先提取并保存了自己硬盘中分区表的信息和DOS分区引导扇区信息，那么，恢复工作变得非常简单。可以直接用Debug将这两种引导扇区的内容分别调入内存，然后分别回它的原来位置，这样就消除了计算机病毒。

2. 文件型计算机病毒

文件型计算机病毒程序都是依附在系统可执行文件或覆盖文件上的，当文件装入系统执行的时候，引导计算机病毒程序也进入系统中。只有极少计算机病毒程序感染数据文件。

此类病毒感染的对象大多是系统的可执行文件，也有一些还要对覆盖文件进行传染，而对数据进行传染的则少见。

清除的方法为：

(1) 确定计算机病毒程序的位置，是驻留在文件尾部还是在文件首部。

(2) 找到计算机病毒程序首部的位置(对应于文件尾部驻留方式)，或者尾部位置(对应于文件首部驻留方式)。

(3) 恢复原文件头部的参数。

(4) 修改文件长度，将源文件写回。

3. 脚本型计算机病毒

主要采用脚本语言设计的病毒称为脚本病毒。实际上在早期的系统中，计算机病毒就已经开始利用脚本进行传播和破坏，不过专门的脚本病毒并不常见。但是在脚本应用无所不在的今天，脚本病毒却成为危害最大、最为广泛的病毒，特别是当它和一些传统的恶性病毒相结合时，其危害就更为严重了。

其主要有两种类型，纯脚本型和混合型。

它的特点是：编写简单、破坏力大、感染力强、传播范围大(多通过E-mail，局域网共享，感染网页文件的方式传播)、计算机病毒源码容易被获取、变种多、欺骗性强。

清除的方法为：

(1) 禁用文件系统对象FileSystemObject。

(2) 卸载Windows Scripting Host。

(3) 删除vbs，vbe，js，jse文件后缀与应用程序映射。

(4) 在Windows目录中，找到WScript.exe，更改名称或者删除。

(5) 要彻底防止vbs网络蠕虫病毒，还需要设置一下浏览器。

(6) 禁止OE的自动收发电子邮件功能。

(7) 显示所有文件类型的扩展名称。

(8) 将系统网络连接的安全级别至少设置为“中等”。

4. 特洛伊木马计算机病毒

特洛伊木马也叫黑客程序或后门病毒，是指经常在正常程序中的一段具有特殊功能的

程序，其隐蔽性极好，不易察觉，是一种极为危险的网络攻击手段。

第一代：伪装性病毒，第二代：AIDS 型木马，第三代：网络传播性木马。

检测的方法包括：

(1) 检查注册表；

(2) 检查系统配置文件。

清除的方法为：

(1) 备份重要数据；

(2) 立即关闭电源；

(3) 备份木马入侵现场；

(4) 修复木马危害。

5. 蠕虫计算机病毒

蠕虫是一种通过网络传播的恶性计算机病毒，它具有计算机病毒的一些共性，如传播性、隐蔽性、破坏性等。同时有自己的一些特征，如利用文件寄生，对网络造成拒绝服务以及和黑客技术相结合等。

简单点说，蠕虫就是使用危害的代码来攻击网络上的受害主机，并在受害主机上自我复制，再攻击其他受害主机的计算机病毒。

其特征是：自我繁殖、利用软件漏洞、造成网络拥堵、消耗系统资源、留下安全隐患。

清除的方法为：

(1) 与防火墙互动；

(2) 交换机联动；

(3) 通知 HIDS(基于主机的入侵检测)。

10.3 计算机木马概述

10.3.1 木马的特性

木马的全称是"特洛伊木马"，是一种新型的计算机网络病毒程序，是一种基于远程控制的黑客工具，它利用自身具有的植入功能，或依附具有传播功能的病毒，进驻目标机器监听、修改、窃取文件。木马有以下 6 个基本特征。

1. 隐蔽性是其首要的特征

当用户执行正常程序时，在难以察觉的情况下，完成危害安全的操作，具有隐蔽性。它的隐蔽性主要体现在 6 个方面：①不产生图标、②文件隐藏、③在专用文件夹中隐藏、④自动在任务管理器中隐形、⑤无声无息地启动、⑥伪装成驱动程序及动态链接库。

2. 具有自动运行性

它是一个当系统启动时即自动运行的程序，所以它必需潜入启动配置文件中，如 win.ini、system.ini、winstart.bat 以及启动组等文件之中。

3. 木马程序具有欺骗性

木马程序要达到其长期隐蔽的目的，就必须借助系统中已有的文件，以防被发现，它经常使用的是常见的文件名或扩展名，如 dll\win\sys\explorer 等字样，或者仿制一些不易被人区分的文件名，如字母 l 与数字 1、字母 o 与数字 0，常修改基本文件中的这些难以分辨的字符，更有甚者干脆借用系统文件中已有的文件名，只不过保存在不同路径之中。还有的木马程序为了隐藏自己，也常把自己设置成一个 ZIP 文件式图标，当用户一不小心打开它时，它就马上运行等手段。那些编制木马程序的人还在不断地研究、发掘新的木马，总之是越来越隐蔽，越来越专业，所以有人称木马程序为"骗子程序"。

4. 具备自动恢复功能

现在很多木马程序中的功能模块已不再是由单一的文件组成的，而是具有多重备份，可以相互恢复。

5. 能自动打开端口

借助服务器客户端的通信手段，利用 TCP/IP 协议不常用的端口自动进行连接，开方便之"门"。

6. 功能的特殊性

通常的木马功能都是十分特殊的，除了普通的文件操作以外，还有些木马具有搜索 cache 中的口令、设置口令、扫描目标机器人的 IP 地址、进行键盘记录、远程注册表的操作，以及锁定鼠标等功能，上面所讲的远程控制软件的功能当然不会有，毕竟远程控制软件是用来控制远程机器，方便自己操作而已，而不是用来黑对方的机器的。

木马的传播途径有：

(1) 利用操作系统和浏览器漏洞传播。

(2) 利用移动存储设备(U 盘)等来传播。

(3) 利用第三方软件(如 Realplayer，迅雷，暴风影音等)漏洞传播。

(4) 利用 ARP 欺骗方式来传播。

(5) 利用电子邮件、QQ、MSN 等通信软件传播。

(6) 利用网页挂马，嵌入恶意代码来传播。

木马病毒的危害有：

(1) 利用通信软件盗取用户的个人信息。黑客可以利用木马病毒盗取用户的如 QQ、MSN 等账号以盗取用户好友的个人信息等。

(2) 盗取网游账号，威胁虚拟财产安全。黑客利用木马病毒盗取用户游戏账户密码，并将用户游戏中的装备或游戏币转移，造成损失。

(3) 盗取用户的网银信息，威胁财产安全。黑客利用木马，采用键盘记录等方法盗取用户的个人银行信息，直接导致用户的经济损失。

(4) 给电脑打开后门，使电脑可能被黑客控制。目前，木马病毒结合了传统病毒的破坏性，产生了更有危害性的混合型木马病毒。有关报告显示：木马病毒占总病毒数的 60%以

上。其中，盗号木马占总木马数的70%以上。从数据上可以看出，木马数量的成倍增长，变种层出不穷，使得计算机用户的处境更加危险。

计算机世界的特洛伊木马(Trojan)是指隐藏在正常程序中的一段具有特殊功能的恶意代码，是具备破坏和删除文件、发送密码、记录键盘和攻击DoS等特殊功能的后门程序。

1）第一代木马：伪装型病毒

这种病毒通过伪装成一个合法性程序诱骗用户上当。世界上第一个计算机木马是出现在1986年的PC-Write木马。它伪装成共享软件PC-Write的2.72版本(事实上，编写PC-Write的Quicksoft公司从未发行过2.72版本)，一旦用户信以为真运行该木马程序，那么他的下场就是硬盘被格式化。在笔者刚刚上大学的时候，曾听说本校一个前辈牛人在WAX机房上用BASIC做了一个登录界面木马程序，当把用户ID、密码输入一个和正常的登录界面一模一样的伪登录界面后，木马程序一边保存该ID和密码，一边提示密码错误让用户重新输入，当用户第二次登录时，已成了木马的牺牲品。此时的第一代木马还不具备传染特征。

2）第二代木马：AIDS型木马

继PC-Write之后，1989年出现了AIDS木马。由于当时很少有人使用电子邮件，所以AIDS的作者就利用现实生活中的邮件进行散播：给其他人寄去一封封含有木马程序软盘的邮件。之所以叫这个名称是因为软盘中含有AIDS和HIV疾病的药品、价格、预防措施等相关信息。软盘中的木马程序在运行后，虽然不会破坏数据，但是它将硬盘加密锁死，然后提示受感染用户花钱消灾。可以说第二代木马已具备了传播特征(尽管通过传统的邮递方式)。

3）第三代木马：网络传播性木马

随着Internet的普及，这一代木马兼备伪装和传播两种特征并结合TCP/IP网络技术四处泛滥。同时还有了三个新特征，第一，添加了“后门”功能；第二，添加了按键记录功能；第三，有了视频监控和桌面监控等功能。

木马的种类可以划分成破坏型、密码发送型、远程访问型、键盘记录木马、DoS攻击木马、代理木马、FTP木马，以及反弹端口型木马，如表10-1所示。

表10-1 木马种类

种　类	特　性	传播途径
破坏型	唯一的功能就是破坏并且删除文件，可以自动删除电脑上的DLL、INI、EXE文件	硬盘传播
密码发送型	向密码输入窗口发送WM_SETTEXT消息模拟输入密码，向按钮窗口发送WM_COMMAND消息模拟单击。在破解过程中，把密码保存在一个文件中，以便在下一个序列的密码再次进行穷举或多部机器同时进行分工穷举，直到找到密码为止	可以找到隐藏密码并把它们发送到指定的信箱。也有些黑客软件长期潜伏，记录操作者的键盘操作，从中寻找有用的密码
远程访问型	最广泛的是特洛伊马，只要有人运行了服务端程序，如果客户知道了服务端的IP地址，就可以实现远程控制	通过控制Internet的UDP进行传播

续表

种　　类	特　　性	传播途径
键盘记录木马	这种特洛伊木马是非常简单的。它们只做一件事情，就是记录受害者的按键情况并且在LOG文件里查找密码	潜伏在计算机硬盘中，通过记录使用者的键盘操作进行传播
DoS攻击木马	随着DoS攻击越来越广泛地应用，被用作DoS攻击的木马也越来越流行。当你入侵了一台机器，给他种上DoS攻击木马，你控制的机器（"肉鸡"）数量越多，发动DoS攻击取得成功的机率就越大。	通过邮件传播，一旦机器被感染，木马就会随机生成各种各样主题的信件，对特定的邮箱不停地发送邮件，一直到对方瘫痪、不能接收邮件为止
代理木马	"代理木马"具有自动下载木马病毒的功能，一旦感染系统后，当系统接入互联网，再从指定的网址下载其他木马、病毒等恶意软件	它们可以根据病毒编者指定的网址下载木马病毒或其他恶意软件，还可以通过网络和移动存储介质传播
FTP木马	这种木马可能是最简单和古老的木马了，它的唯一功能就是打开21端口，等待用户连接	控制用户的21端口使其运行某一指定的命令
反弹端口型木马	木马定时监测控制端的存在，发现控制端上线立即弹出端口主动连接控制端打开的主动端口；即使用户使用扫描软件检查自己的端口，发现类似TCP的情况	通过控制计算机防火墙端口进行传播

10.3.2　计算机木马的原理

木马病毒通常包括两个部分：服务器和客户端。服务端植入危害主机，而施种者利用客户端侵入运行了服务端的主机。木马的服务端一旦启动，受害主机的一个或几个端口即对施种者敞开，使得攻击者可以利用这些端口对受害主机执行入侵操作。

基本的木马植入过程包括：

1. 配置、传播木马

一般来说一个设计成熟的木马都有木马配置程序，从具体的配置内容看，主要是为了实现伪装和信息反馈两方面的功能。

木马的传播方式主要有两种：一种是通过E-mail，控制端将木马程序以附件的形式夹在邮件中发送出去，收信人只要打开附件系统就会感染木马；另一种是软件下载，一些非正规的网站以提供软件下载为名，将木马捆绑在软件安装程序上，下载后，只要一运行这些程序，木马就会自动安装。

2. 运行木马

服务端用户运行木马或捆绑木马的程序后，木马就会自动进行安装。首先将自身拷贝到Windows的系统文件夹中（C:\Windows或C:\Windows\System目录下），然后在注册表、启动组、非启动组中设置好木马的触发条件，这样木马的安装就完成了。

木马被激活后，进入内存，并开启事先定义的木马端口，准备与控制端建立连接。这时服务端用户可以在MS-DOS方式下，输入NETSTAT -AN查看端口状态，一般个人电脑在

脱机状态下是不会有端口开放的，如果有端口开放，就要注意是否感染木马了。

3. 信息反馈

木马配置程序将就信息反馈的方式或地址进行设置，如设置信息反馈的邮件地址、IRC号、ICO号等。

4. 建立连接

一个木马连接的建立首先必须满足两个条件：一是服务端已安装了木马程序；二是控制端，服务端都要在线。在此基础上控制端可以通过木马端口与服务端建立连接。值得一提的是，要扫描整个IP地址段显然费时费力，一般来说控制端都是先通过信息反馈获得服务端的IP地址，由于拨号上网的IP是动态的，即用户每次上网的IP都是不同的。

5. 远程控制

木马连接建立后，控制端端口和木马端口之间将会出现一条通道，控制端上的控制端程序可利用这条通道与服务端上的木马程序取得联系，并通过木马程序对服务端进行远程控制。

6. 木马病毒的传播及植入

由于木马病毒是一种非自我复制的恶意代码，因此它需要依靠用户向其他人发送其拷贝。木马病毒可以作为电子邮件附件或者隐藏在用户与其他用户进行交互的文档或者其他文件中。它们还可以被其他恶意代码所携带，如蠕虫。木马病毒有时也会隐藏在从互联网上下载的捆绑软件中。当用户安装此软件时，病毒就会在后台秘密安装。木马植入技术主要是指木马病毒利用各自的途径进入目标机器的具体方法。

7. 木马病毒植入技术

木马病毒植入技术，主要是指木马病毒利用各种途径进入目标机器的具体实现方法。

（1）利用电子邮件进行传播：攻击者将木马程序伪装成E-mail附件的形式发送过去，收信方只要查看邮件附件就会使木马程序得到运行并安装进入系统。

（2）利用网络下载进行传播：一些非正规的网站以提供软件下载为名，将木马捆绑在软件安装程序上，下载后，只要运行这些程序，木马就会自动安装。

（3）利用网页浏览传播：这种方法利用Java Applet编写出一个HTML网页，当用户浏览该页面时，JavaApplet会在后台将木马程序下载到计算机缓存中，然后修改注册表，使指向木马程序。

（4）利用一些漏洞进行传播：如微软著名的IIS服务器溢出漏洞，通过一个IIS HACK攻击程序即可使IIS服务器崩溃，并且同时在受控服务器执行木马程序。由于微软的浏览器在执行Script脚本上存在一些漏洞，攻击者可以利用这些漏洞传播病毒和木马，甚至直接对浏览者主机进行文件操作等控制。

（5）远程入侵进行传播：黑客通过破解密码和建立IPC远程连接后登录到目标主机，将木马服务端程序拷贝到计算机中的文件夹（一般在C:\Windows\system32或者C:\

WINNT\sys-tem32)中,然后通过远程操作让木马程序在某一个时间运行。

(6) 基于DLL和远程线程插入的木马植入:这种传播技术是以DLL的形式实现木马程序,然后在目标主机中选择特定目标进程(如系统文件或某个正常运行的程序),由该进程将木马DLL植入本系统中的。

(7) 利用蠕虫病毒传播木马:网络蠕虫病毒具有很强的传染性和自我复制能力,将木马和蠕虫病毒结合在一起就可以大大地提高木马的传播能力。结合了蠕虫病毒的木马利用病毒的特性,在网络上进行传播、复制,这就加快了木马的传播速度。

当木马病毒成功植入目标机后,就必须确保自己可以通过某种方式得到自动运行。常见的木马病毒加载技术主要包括系统启动自动加载、文件关联和文件劫持等。

系统启动自动加载,是最常见的木马自动加载方法。木马病毒通过将自己复制到启动组,或在win.ini,system.ini和注册表中添加相应的启动信息而实现系统启动时自动加载。这种加载方式简单有效,但隐蔽性差。目前很多反木马软件都会扫描注册表的启动键(信息),故而新一代木马病毒都采用了更加隐蔽的加载方式。

文件劫持,是一种特殊的木马加载方式。木马病毒被植入目标机后,需要首先对某个系统文件进行替换或嵌入操作,使得该系统文件在获得访问权之前,木马病毒被率先执行,然后再将控制权交还给相应的系统文件。采用这种方式加载木马不需要修改注册表,从而可以有效地躲过注册表扫描型反木马软件的查杀。这种方式最简单的实现方法是将某系统文件改名,然后将木马程序改名。这样当这个系统文件被调用的时候,实际上是木马程序被运行,而木马启动后,再调用相应的系统文件并传递原参数。

木马的防范可以从下面几个方面着手:

(1) 不随意打开来历不明的邮件,阻塞可疑邮件。

(2) 不随意下载来历不明的软件。

(3) 及时修补漏洞和关闭可疑的端口。

(4) 尽量少用共享文件夹。

(5) 运行实时监控系统。

(6) 经常升级系统和更新杀毒软件。

(7) 限制不必要的具有传输能力的文件。

(8) 关闭不常使用的端口。

习题

1. 什么是计算机病毒?
2. 计算机病毒的类型有哪些?
3. 计算机病毒有什么危害?
4. 计算机木马的威胁有哪些?

第11章 入侵检测技术

11.1 入侵检测概述

1980年,詹姆斯·安德森(James P. Anderson)第一次系统阐述了入侵检测的概念,并将入侵行为分为外部渗透、内部渗透和不法行为三种,还提出了利用审计数据监视入侵活动的思想。

Anderson将入侵尝试或威胁定义为:潜在的、有预谋的、未经授权的访问信息、操作信息、致使系统不可靠或无法使用的企图。而入侵检测的定义为:发现非授权使用计算机的个体(如“黑客”)或计算机系统的合法用户滥用其访问系统的权利以及企图实施上述行为的个体。执行入侵检测任务的程序即入侵检测系统。入侵检测系统也可以定义为:检测企图破坏计算机资源的完整性,真实性和可用性行为的软件。

入侵检测系统执行的主要任务包括:监视、分析用户及系统活动;审计系统构造和弱点;识别、反映已知进攻的活动模式,向相关人士报警;统计分析异常行为模式;评估重要系统和数据文件的完整性;审计、跟踪管理操作系统,识别用户违反安全策略的行为。入侵检测一般分为三个步骤:信息收集、数据分析、响应。

入侵检测的目的:①识别入侵者;②识别入侵行为;③检测和监视已实施的入侵行为;④为对抗入侵提供信息,阻止入侵的发生和事态的扩大。

11.2 入侵检测的系统结构和分类

11.2.1 入侵检测的系统结构

入侵检测系统的基本结构构成公共入侵检测框架(Common Intrusion Detection Frame,CIDF)提出了通用模型,将入侵检测系统分为4个基本组件:事件产生器、事件分析器、响应单元和事件数据库,如图11-1所示。

1. 事件产生器

事件产生器(Event Generators)是入侵检测系统中负责原始数据采集的部分,它对数据流、日志文件等进行追踪,然后将收集到的原始数据转换为事件,并向系统的其他部分提

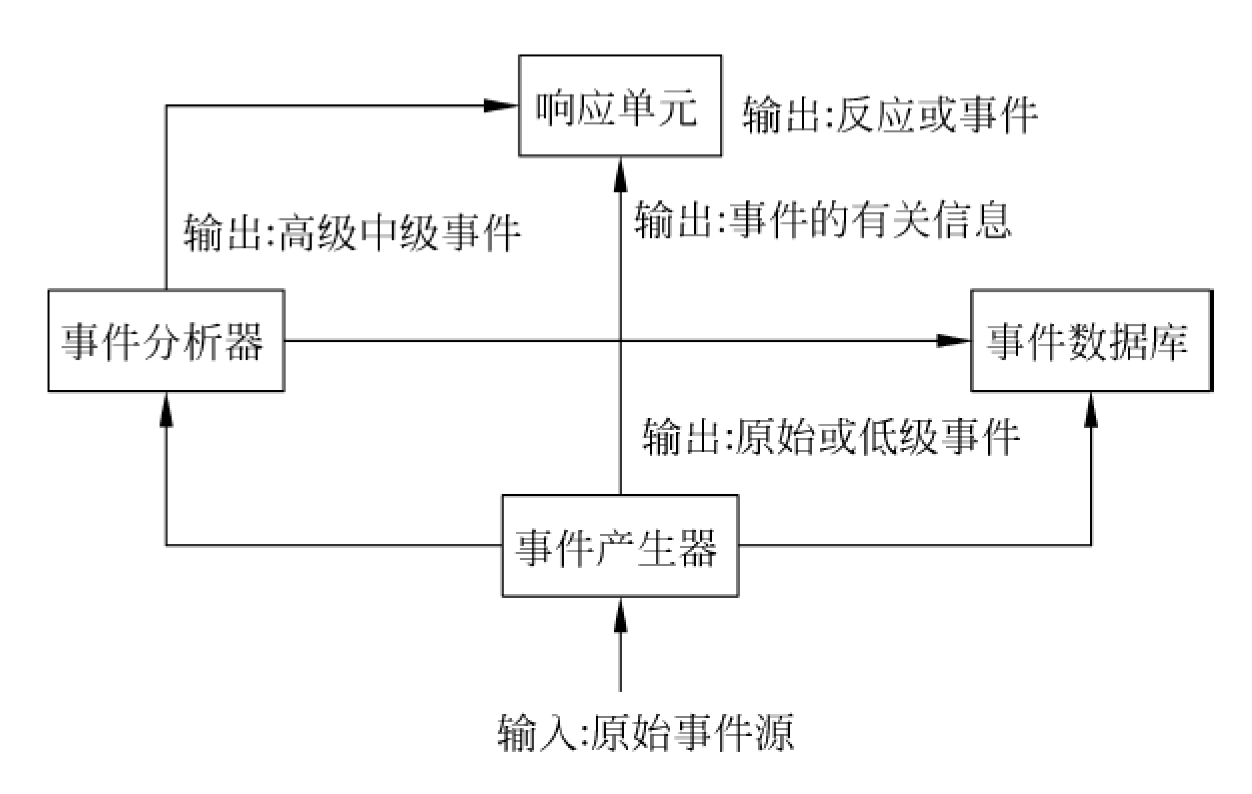


图 11-1　入侵检测的体系结构

供此事件。

2. 事件分析器

事件分析器(Event Analyzers) 接收事件信息,然后对它们进行分析,判断是不是入侵行为或异常现象,最后把判断的结果转换为警告信息。

3. 事件数据库

事件数据库(Response Units)是存放各种中间和最终数据的地方。

4. 响应单元

响应单元(Response Units)根据警告信息做出反应,如切断连接、改变文本属性等强烈的反应,也可能是简单地报警。它是入侵检测系统中的主动武器。

11.2.2　入侵检测的分类

通过对现有的入侵检测系统和入侵检测技术的研究,可以从以下几个方面对入侵检测系统进行分类。

1. 根据目标系统类型的不同划分

(1) 基于主机(host-based)的入侵检测系统。通常,基于主机的入侵检测系统可以检测系统、事件和操作系统下的安全记录以及系统记录。当文件发生变化时,入侵检测系统将新的记录条目与攻击标记相比较,看它们是否匹配。如果匹配,系统就会向管理员报警,以采取措施。基于主机的入侵检测系统如图 11-2 所示。

(2) 基于网络(network-based)的入侵检测系统基于网络的入侵检测系统使用原始网络数据包作为数据源。它通常利用一个运行在混杂模式下的网络适配器来实时监视并分析网络的所有通信业务。基于网络的入侵检测系统的基本结构如图 11-3 所示。

2. 根据入侵检测分析方法的不同划分

(1) 误用入侵检测(特征检测 signature-based)。误用检测是基于已知的系统缺陷和入

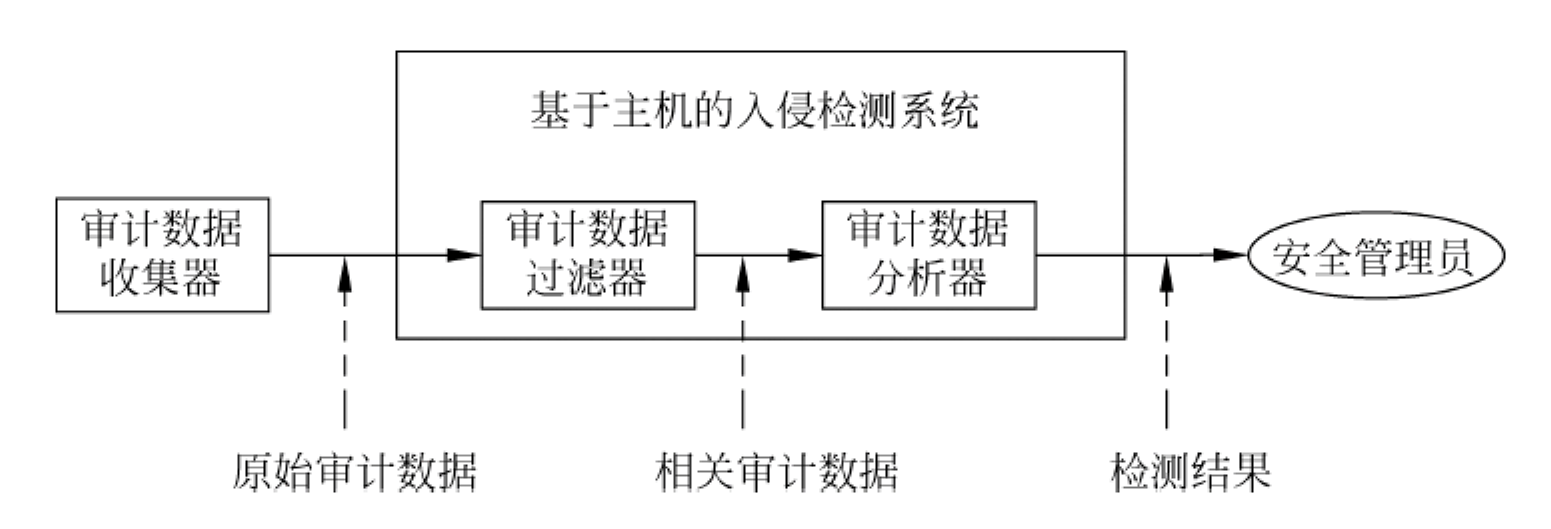


图 11-2 基于主机的入侵检测系统的基本结构

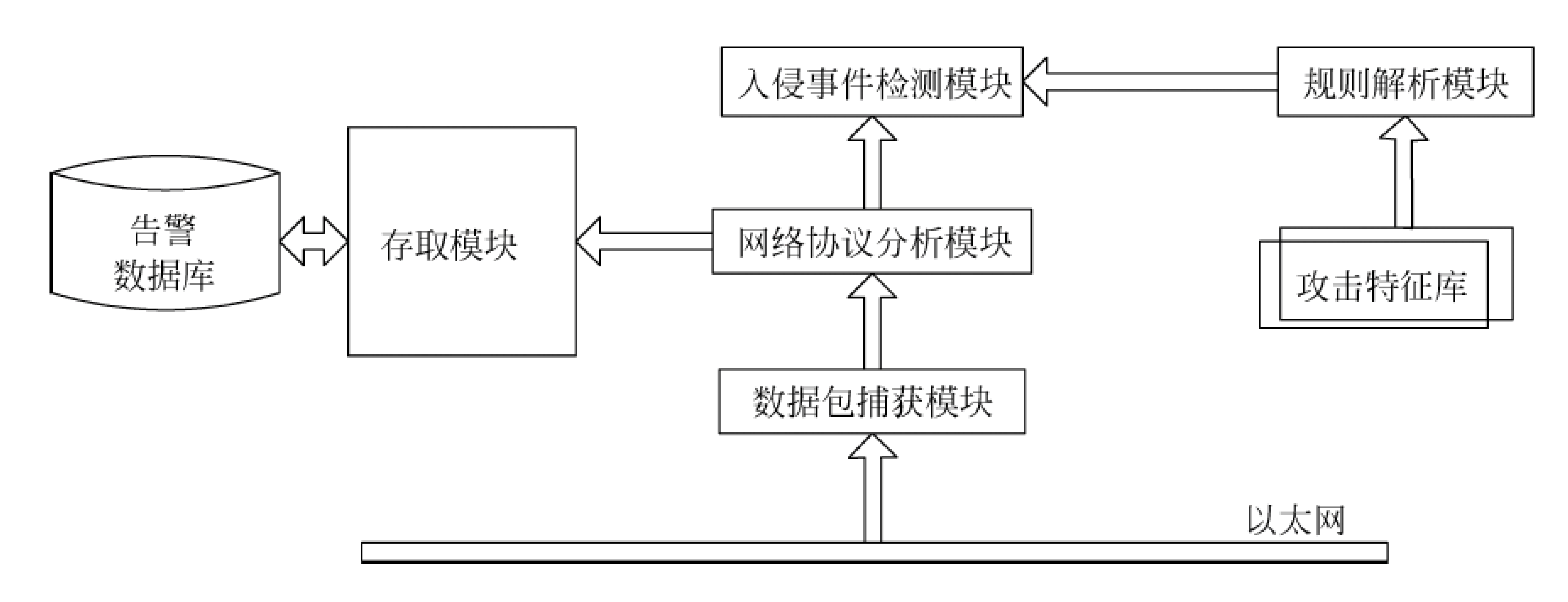


图 11-3 基于网络的入侵检测系统的基本结构

侵模式，所以又称为特征检测。误用检测是对不正常的行为建模，这些不正常的行为是被记录下来的确认的误用和攻击。通过对系统活动的分析，发现与被定义好的攻击特征相匹配的事件或事件集合。该检测方法可以有效地检测到已知攻击，检测精度高，误报少。但需要不断更新攻击的特征库，系统灵活性和自适应性较差，漏报较多。商用 IDS 多采用该种检测方法。

(2) 异常入侵检测(anomaly-based)。异常检测是指能根据异常行为和使用计算机资源的情况检测出入侵的方法。它试图用定量的方式描述可以接受的行为特征，以区分非正常的、潜在的入侵行为。也就是，异常检测是对用户的正常行为建模，通过正常行为与用户的行为进行比较，如果二者的偏差超过了规定阈值则认为该用户的行为是异常的。异常检测的误报较多。目前，大多数异常检测技术还处于研究阶段，基本没有用于商业 IDS 中。

3. 根据检测系统各个模块运行分布方式的不同划分

(1) 集中式入侵检测系统。集中式 IDS 有多个分布在不同主机上的审计程序，仅有一个中央入侵检测服务器。审计程序将当地收集到的数据踪迹发送给中央服务器进行分析处理。随着服务器所承载的主机数量的增多，中央服务器进行分析处理的数量就会猛增，而且一旦服务器遭受攻击，整个系统就会崩溃。

(2) 分布式(协作式)入侵检测系统。分布式 IDS 是将中央检测服务器的任务分配给多个基于主机的 IDS，这些 IDS 不分等级，各司其职，负责监控当地主机的某些活动。所以，其可伸缩性、安全性都得到了显著的提高，并且与集中式 IDS 相比，分布式 IDS 对基于网络的共享数据量的要求较低。但维护成本却提高了很多，并且增加了所监控主机的工作负荷，如通信机制、审计开销、踪迹分析等。

4. 根据入侵检测系统分析的数据来源不同划分

根据入侵检测系统分析的数据来源分为主机系统日志、原始的网络数据包、应用程序的日志、防火墙报警日志以及其他入侵检测系统的报警信息等。据此可将入侵检测系统分为基于不同分析数据源的入侵检测系统。

5. 根据系统对入侵攻击的响应方式分类

(1) 主动的入侵检测系统系统。在检测出入侵后,可自动地对目标系统中的漏洞采取修补、强制可疑用户(可能的入侵者)退出以及关闭相关服务等对策和相应措施。

(2) 被动的入侵检测系统。检测出对系统的入侵攻击后只是产生报警信息通知系统安全管理员,至于之后的处理工作则由系统管理员来完成。

11.3 入侵检测的功能

入侵检测系统的主要功能为:

(1) 监视并分析用户和系统的行为;

(2) 审计系统配置和漏洞;

(3) 评估敏感系统和数据的完整性;

(4) 识别攻击行为、对异常行为进行统计;

(5) 自动收集与系统相关的补丁;

(6) 审计、识别、跟踪违反安全法规的行为;

(7) 使用诱骗服务器记录黑客行为。

11.3.1 入侵检测系统的功能结构

入侵检测是防火墙的合理补充,帮助系统对付来自外部或内部的攻击,扩展了系统管理员的安全管理能力(如安全审计、监视、攻击识别及其响应),提高了信息安全基础结构的完整性。

入侵检测系统的主要工作就是从信息系统的若干关键点上收集信息,然后分析这些信息,用来获悉网络中有无违反安全策略的行为和遭到袭击的迹象。

无论对于什么类型的入侵检测系统,其工作模式都可以体现为以下步骤,如图 11-4 所示。

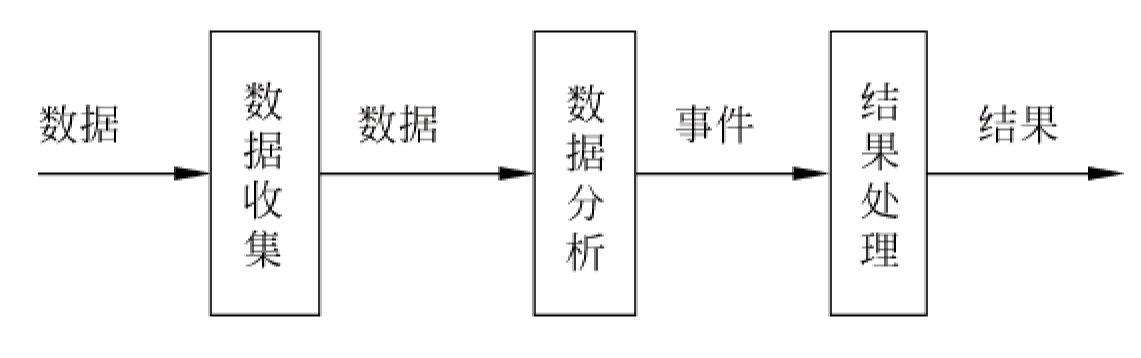


图 11-4 工作模式

一个典型的入侵检测系统从功能上可以分为三个组成部分:感应器(Sensor)、分析器(Analyzer)和管理器(Menager)。

1. 感应器

感应器负责收集信息。

(1) 收集的数据内容。主机和网络日志文件；目录和文件中不期望的改变；程序执行中的不期望行为；物理形式的入侵信息。

(2) 入侵检测系统的数据收集机制。基于主机的数据收集和基于网络的数据收集；分布式与集中式数据收集机制；直接监控和间接监控；外部探测器和内部探测器。

2. 分析器

分析器从许多感应器接收信息，并对这些信息进行分析以决定是否有入侵行为发生，就是对从数据源提供的系统运行状态和活动记录进行同步、整理、组织、分类以及各种类型的细致分析，提取其中包含的系统活动特征或模式，用于对正常和异常行为的判断。

3. 管理器

管理器通常也被称为用户控制台，它以一种可视的方式向用户提供收集到的各种数据及相应的分析结果，用户可以通过管理器对入侵检测系统进行配置，设定各种系统的参数，从而对入侵行为进行检测以及相应的措施进行管理。

一个好的 IDS 应该让用户能够裁剪定制其响应机制，以符合特定的需求环境。

(1) 主动响应。系统自动或以用户设置的方式阻断攻击过程或以其他方式影响攻击过程，通常可以选择的措施有：针对入侵者采取的措施；修正系统；收集更详细的信息。

(2) 被动响应。在被动响应系统中，系统只报告和记录发生的事件。

11.3.2 入侵检测系统的部署

对于入侵检测系统来说，其类型不同，应用环境不同，部署方案也就会有所差别。

1. 在基于网络的 IDS 中部署入侵检测系统

基于网络的 IDS 主要检测网络数据报文，因此一般将检测系统部署在靠近防火墙的地方。具体可安排在图 11-5 中的几个位置。

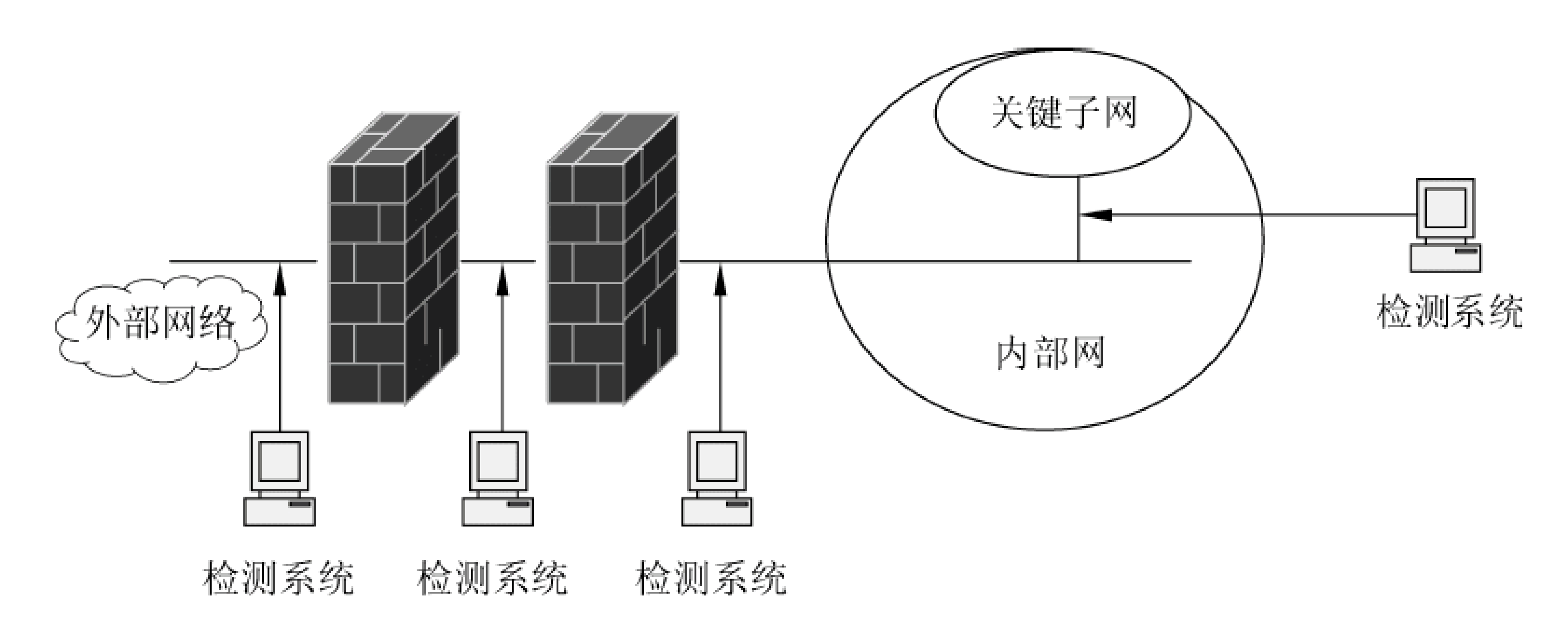


图 11-5 在基于网络的 IDS 中部署入侵检测器

其中，检测器可安放的位置有：隔离区（DeMilitarized Zone，DMZ）也称"非军事化区"；内网主干（防火墙内侧）；外网入口（防火墙外侧）；在防火墙的内外都放置；关键子网。

2. 在基于主机的 IDS 中部署入侵检测器

基于主机的 IDS 通常是一个程序，部署在最重要、最需要保护的主机上用于保护关键主机或服务器。

11.4　Windows 下入侵检测系统的设计

Windows 平台下基于规则（基于误用）的网络入侵检测系统，在现有已知的入侵特征下建立规则库，实现数据包的捕获和分析，完成对漏洞攻击和扫描等攻击行为的检测和报告。同时将攻击事件存入数据库，以便事后取证。

系统基于 Windows 平台构建的基于规则（基于误用）的网络入侵检测系统，能够有效检测入侵事件、防止入侵、安全审计。

系统总体结构框架如图 11-6 所示。

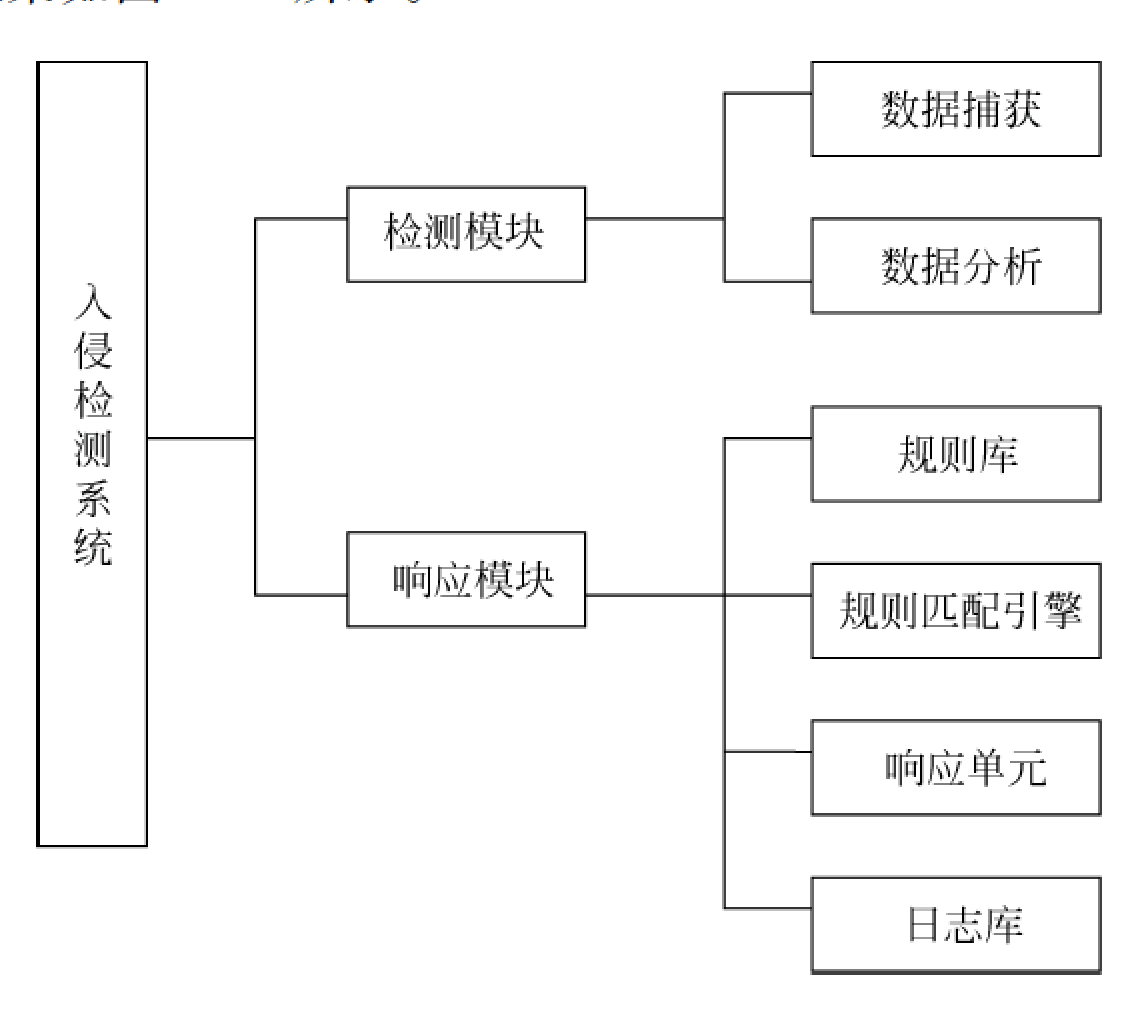


图 11-6　系统总体结构框架

在规则库的设计中，IDS 要有效地捕捉入侵行为，必须拥有一个强大的入侵特征数据库，这就如同公安部门必须拥有健全的罪犯信息库一样。但是，IDS 一般所带的特征数据库都比较死板，遇到"变脸"的入侵行为往往相逢不相识。因此，管理员有必要学会如何创建满足实际需要的特征数据样板，做到万变应万变。

IDS 中的规则（特征）就是指用于判别通信信息种类的样板数据，通常分为多种，以下是一些典型情况及识别方法。

（1）来自保留 IP 地址的连接企图：可通过检查 IP 报头（IP Header）的来源地址轻易地识别。

（2）带有非法 TCP 标识联合物的数据包：可通过对比 TCP 报头中的标识集与已知正确和错误标记联合的不同点来识别。

(3) 含有特殊病毒信息的 E-mail：可通过对比每封 E-mail 的主题信息和病态 E-mail 的主题信息来识别，或者，通过搜索特定名字的附近来识别。

(4) 查询负载中的 DNS 缓冲区溢出企图：可通过解析 DNS 域及检查每个域的长度来识别利用 DNS 域的缓冲区溢出企图。

(5) 通过对 POP3 服务器发出上千次同一命令而导致的 DOS 攻击：通过跟踪记录某个命令连续发出的次数，看看是否超过了预设上限，而发出报警信息。

(6) 未登录情况下使用文件和目录命令对 FTP 服务器的文件访问攻击：通过创建具备状态跟踪的特征样板以监视成功登录的 FTP 对话、发现未经验证却发命令的入侵企图。

从以上分类可以看出特征的涵盖范围很广，有简单的报头域数值、有高度复杂的连接状态跟踪、有扩展的协议分析。

报头值的结构比较简单，而且可以很清楚地识别出异常报头信息，因此，特征数据的首席候选人就是它。一个经典的例子是，明显违背 RFC793 中规定的 TCP 标准、设置了 SYN 和 FIN 标记的 TCP 数据包。这种数据包被许多入侵软件采用，向防火墙、路由器以及 IDS 系统发起攻击。

因为大多数操作系统和应用软件都是在假定 RFC 被严格遵守的情况下编写的，没有添加针对异常数据的错误处理程序，所以许多包含报头值的漏洞都会故意违反 RFC 的标准定义，明目张胆地揭发被攻击对象的偷工减料行为。许多包含错误代码的不完善软件也会产生违反 RFC 定义的报头值数据。并非所有的操作系统和应用程序都能全面拥护 RFC 定义，至少会存在一个方面与 RFC 不协调。随着时间的推移，执行新功能的协议可能不被包含于现有 RFC 中。由于以上几种情况，严格基于 RFC 的 IDS 特征数据就有可能产生漏报或误报效果。对此，RFC 也随着新出现的违反信息而不断进行着更新，有必要定期地回顾或更新存在的特征数据定义。非法报头值是特征数据一个非常基础的部分，合法但可疑的报头值也同等重要。例如，如果存在到端口 31337 或 27374 的可疑连接，就可报警说可能有特洛伊木马在活动；再附加上其他更详细的探测信息，就能够进一步判断是真马还是假马。

不同的入侵检测系统有不同的规则库，本系统的规则库结构如表 11-1 所示。

表 11-1　规则库结构

选项名称	数据库定义名称	意　义
规则号	rulesid	用于定义规则的编号
IP 协议字段值	Ip_proto	在规则中上层协议的代码
IP 上层协议	protocol	IP 上层协议的名称，如 tcp
规则动作	action	决定如果匹配该规则将采取的行动
IP 源地址	ip_src	数据包的来源地址
IP 目的地址	ip_dst	数据包的目的地址
源端口	sourceport	数据包的来源端口
目的端口	destinationport	数据包的去向端口
服务类型	tos	IP 数据包中 TOS 字段的值
存活期	ttl	设置一个用于检查的存活期值
IP 头的分片 id	id	用于检测 IP 头的分片 id 值
IP 选项	ipoption	如果 IP 包头中包含则检查，如源路由
IP 分段位	fragbits	IP 头中分段位和保留位的值

续表

选 项 名 称	数据库定义名称	意　　义
数据大小	datasize	数据包净荷大小
TCP Flags	flags	检查 TCP Flags 的值
TCP 序号	seq	检查 TCP 顺序号的值
TCP 应答	ack	检查 TCP 应答的值
TCP 窗口域	window	测试 TCP 窗口域的特殊值
ICMP 类型	itype	检查 ICMP type 的值
ICMP 代码	icode	检查 ICMP code 的值
ICMP ECHO ID	icmp_id	检查 ICMP ECHO ID 的值
ICMP ECHO 顺序号	icmp_seq	检查 ICMP ECHO 顺序号的值
包净荷中搜索指定的匹配串	content	需要检测的数据段的匹配字符串
包净荷中开始搜索的位置	offset	搜索数据段的开始位置
包净荷中搜索的最大深度	depth	数据段从开始位置开始搜索的最大长度
字符串大小	nocase	搜索的字符串是否考虑大小写
警告信息	msg	需要输出或写入日志数据库的警告信息

规则在数据库中是以所有类型混合存储的方式存储的，所以如果每个数据包到来时都遍历数据库，将消耗系统的资源和增加程序的时间复杂度，所以在本入侵检测系统中将建立规则的动态链表，在程序初始化时读取数据库，将以不同的协议类型，加入不同的动态规则链表。规则动态链表的分类如表 11-2 所示。

表 11-2　规则链表分类

IP 规则动态链表	以数据库规则记录中的字段为 IP 时，加入该链表
TCP 规则动态链表	以数据库规则记录中的字段为 TCP 时，加入该链表
UDP 规则动态链表	以数据库规则记录中的字段为 UDP 时，加入该链表
ICMP 规则动态链表	以数据库规则记录中的字段为 ICMP 时，加入该链表

根据数据包的协议特征分为 5 种主要的处理方式：

(1) TCP 规则匹配。如果数据包网络层采用 IP 协议，且 IP 上层协议为 TCP，那么进入 TCPMatch()规则匹配的主函数进行匹配，匹配函数将遍历 TCP 规则动态链表，匹配当前规则中存在的规则选项。

(2) UDP 规则匹配。如果数据包网络层采用 IP 协议，且 IP 上层协议为 UDP，那么进入 UDPMatch()规则匹配的主函数进行匹配，匹配函数将遍历 UDP 规则动态链表，匹配当前规则中存在的规则选项。

(3) ICMP 规则匹配。如果数据包网络层采用 IP 协议，且 IP 上层协议为 ICMP，那么进入 ICMPMatch()规则匹配的主函数进行匹配，匹配函数将遍历 UDP 规则动态链表，匹配当前规则中存在的规则选项。

(4) IP 规则匹配。如果数据包网络层采用 IP 协议，且 IP 上层协议不为 TCP、UDP、ICMP 那么进入 IPMatch()规则匹配的主函数进行匹配，匹配函数将遍历 IP 规则动态链表，匹配当前规则中存在的规则选项。

(5) 其他情况处理。如果数据包网络层采用其他协议，那么根据不同的协议类型输出

相应的信息。

另外,如果规则中出现了数据选项(content),则可以用BM算法对内容进行搜索。内容可以有不同的格式:二进制,文本或者两者的混合。

日志数据库主要是存储需要记录的重要入侵记录,其作用是为了对入侵事件的分析和取证,主要结构如表11-3和表11-4所示。

表11-3　重要日志信息表

名　　称	数据库定义字段名称
协议	protocol
时间	date
警告信息	msg
源和目的IP地址	ip
源和目的端口	port
信息摘要	text

表11-4　一般日志信息表

名　　称	数据库定义字段名称
警告信息	msg
时间	date



习题

1. 根据检测原理,入侵检测系统可以分为几类?其原理分别是什么?
2. 基于主机的入侵检测系统的优点是什么?
3. 基于网络的入侵检测系统的优点与缺点是什么?
4. 入侵检测系统弥补了防火墙的哪些不足?

第12章 无线网络安全技术

12.1 无线网络概述

无线局域网的历史及发展无线局域网的历史起源可以追溯到半个多世纪前的第二次世界大战期间，当时美国陆军采用无线电信号用作资料传输。他们研发出一套无线电传输技术，并且采用了相当高强度的加密技术，美军和盟军都广泛使用这项技术。1971 年，夏威夷大学的研究人员从美军在第二次世界大战时期应用的这项技术中得到灵感，创造了第一个基于封包式技术的无线电通信网络。这个被称做 ALOHNET 的网络包括 7 台计算机，它们采用双向星型拓扑，网络横跨四座夏威夷的岛屿，中心计算机放置在瓦胡岛上。它可以称做无线局域网的鼻祖。

无线局域网可以应用于区域覆盖和点对点传输，其中又以区域覆盖应用占绝大多数。国内的无线局域网现状，按照应用规模大致可以分为以下四种类型。

(1) 个人及家庭用户：拥有多台计算机的家庭越来越普遍，此类用户一般会使用集成无线功能的宽带路由器。

(2) 企业用户：这类用户的无线网络大部分由自己或系统集成商搭建，网络规模差异很大，网络的设计水平和安全状况也参差不齐。

(3) 热点应用：近年来，在咖啡厅，酒店，医院，商场，车站等场所纷纷兴起了提供无线上网的服务，这些网络仍然是由用户自己或系统集成商搭建，但是网络的利用率不高。

(4) 大面积覆盖：为提升城市形象或出于 ISP 之间的竞争等目的，目前涌现了大批机场覆盖、无线社区、无线高校，甚至无线城市，这类大型网络一般是由各大运营商进行部署的，从设计到实施都比较系统。

上面这些不同规模的网络具有各自不同的特点，但它们的共同点就是能够极大地方便人们的生活。不管未来会采用何种无线技术标准，可以肯定的一点是，未来无线网络的传输速率及稳定性将会不断提高，甚至会超越传统有线网络，同时硬件设备的成本将呈下降趋势，结合无线网络移动性强、易扩展、易部署等传统优势，未来无线网络在各个层次各种规模的消费群体里面都将具有极大的发展空间。无线网络比有线网络为用户提供了更大的便携性和灵活性，其能够通过无线接入点(AP)将客户端联接到网络上，从而摆脱电缆的困扰。其接入点可以通过一个具有 RJ-45 接口的网络适配器在有线网络上进行联接。无线网络接入点的典型覆盖范围直径大约为室外 300m 以内，室内 100m 以内，便携计算机等移动设备

可以在其覆盖范围内自由走动,把这些范围连接起来可形成更大的覆盖,可以使用户在多个AP之间移动。

无线网络的出现,使得许多有线网络解决不了的问题迎刃而解。可以在不像传统网络布线的同时,提供有线网络的所有功能,并能够随着用户的需要随意更改扩展网络,实现移动应用。无线网络具有传统有线网络无法比拟的优点。

(1) 灵活性,不受线缆的限制,可以随意增加和配置工作站。

(2) 低成本,无线网络不需要大量的工程布线,同时节省了线路维护的费用。

(3) 移动性,不受时间、空间的限制,随时随地可以上网。

(4) 易安装,和有线相比,无线网络的组建、配置、维护都更容易。

(5) 更加美观,传统的有线网络很多情况下都影响到了家庭的美观,而无线网络则没有这个问题。

但是,一切事物有利亦有弊。无线网络也同时有着许多缺陷。

(1) 无线网络的速度并不是非常稳定,和有线相比,还有着很大的差距。

(2) 安全性也是一个很大的问题,无线网络是通过特定的无线电波传送的。所以在这发射频率的有效范围内,任何具有合适的接收设备的人都可以捕获该频率的信号,而防火墙对通过无线电波进行的网络通信起不了作用,任何人在视距范围之内都可以截获和插入数据。所以无线网络在通信过程中存在着重大的安全威胁。

12.2 无线网络原理

无线局域网,是通过发射和接收装置(无线设备)连接交换机的,工作站通过无线网卡和无线设备进行通信,无线设备接收到信号就传送给交换机再用交换机连接到路由器,路由器接入Internet,实现上网。无线路由器可以直接接入Internet接收无线(广域网比如CDMA)信号来上网。

具体的调制方式包括扩展频谱方式和窄带调制方式。

1. 扩展频谱方式

在这种方式下,数据信号的频谱被扩展成几倍甚至几十倍后再被发射出去。这一做法固然牺牲了频带带宽,但提高了通信系统的抗干扰能力和安全性。采用扩展频谱方式的无线局域网一般选择的是ISM频段,这里ISM分别取自Industrial、Scientific及Medical的第一个字母。许多工业、科研和医疗设备的发射频率均集中于该频段。例如美国ISM频段由902～928MHz,2.4～2.48GHz,5.725～5.850GHz三个频段组成。如果发射功率及带宽辐射满足美国联邦通信委员会(FCC)的要求,则无须向FCC提出专门的申请即可使用ISM频段。

2. 窄带调制方式

顾名思义,在这种调制方式下,数据信号在不做任何扩展的情况下即被直接发射出去。与扩展频谱方式相比,窄带调试方式占用频带少,频带利用率高。但采用窄带调制方式的无线局域网要占用专用频段,因此需经过国家无线电管理部门的批准方可使用。当然,用户也

可以直接选用 ISM 频段来免去频段申请。但所带来的问题是，当临近的仪器设备或通信设备也在使用这一频段时，会严重影响通信质量，通信的可靠性无法得到保障。

目前，基于 IEEE 802.11 标准的 WLAN 使用的均是扩展频谱方式。

迄今为止，电子电器工程师协会(IEEE)已经开发并制定了 4 种 IEEE 802.11 无线局域网规范：IEEE 802.11、IEEE 802.11b、IEEE 802.11a、IEEE 802.11g。所有的这 4 种规范都使用了防数据丢失特征的载波检测多址连接(CDMA/CD)作为路径共享协议。任何局域网应用、网络操作系统以及网络协议(包括互联网协议、TCP/IP)都可以轻松运行在基于 IEEE 802.11 规范的无线局域网上，就像以太网那样。但是 WLAN 却没有“飞檐走壁”的连接线缆。

早期的 IEEE 802.11 标准数据传输率为 2Mb/s，后经过改进，传输速率达 11Mb/s 的 IEEE 802.11b 也紧跟着出台。但随着网络的发展，特别是 IP 语音、视频数据流等高带宽网络应用的频繁，IEEE 802.11b 规范 11Mb/s 的数据传输率不免有些力不从心。于是，传输速率高达 54Mb/s 的 IEEE 802.11a 和 IEEE 802.11g 随即诞生。下面就从性能及特点上入手，来分别介绍这三种当今主流的无线网络规范。

(1) IEEE 802.11b。从性能上看，IEEE 802.11b 的带宽为 11Mb/s，实际传输速率在 5Mb/s 左右，与普通的 10Base-T 规格有线局域网持平。无论是家庭无线组网还是中小企业的内部局域网，IEEE 802.11b 都能基本满足使用要求。由于基于的是开放的 2.4GHz 频段，因此 IEEE 802.11b 的使用无需申请，既可作为对有线网络的补充，又可自行独立组网，灵活性很强。

从工作方式上看，IEEE 802.11b 的运作模式分为两种：点对点模式和基本模式。其中点对点模式是指无线网卡和无线网卡之间的通信方式，即一台装配了无线网卡的计算机可以与另一台装配了无线网卡的计算机实施通信，对于小型无线网络来说，这是一种非常方便的互联方案；而基本模式则是指无线网络的扩充或无线和有线网络并存时的通信方式，这也是 IEEE 802.11b 最常用的连接方式。此时，装载无线网卡的计算机需要通过无线(“接入点”AP)才能与另一台计算机连接，由接入点来负责频段管理及漫游等指挥工作。在带宽允许的情况下，一个接入点最多可支持 1024 个无线节点的接入。当无线节点增加时，网络存取速度会随之变慢，此时添加接入点的数量可以有效地控制和管理频段。从目前大多数的应用案例来看，接入点是作为架起无线网与有线网之间的桥梁而存在的。这一点，在随后的 AP 评测中还将详细阐述。

作为目前最普及、应用最广泛的无线标准，IEEE 802.11b 的优势不言而喻。技术的成熟，使得基于该标准网络产品的成本得到了很好的控制，无论家庭还是企业用户，无需太多的资金投入即可组建一套完整的无线局域网。但 IEEE 802.11b 的缺点也是显而易见的，11Mb/s 的带宽并不能很好地满足大容量数据传输的需要，只能作为有线网络的一种补充。

(2) IEEE 802.11a。就技术角度而言，IEEE 802.11a 与 IEEE 802.11b 虽在编号上仅一字之差，但两者间的关系并不像其他硬件产品换代时的简单升级，这种差别主要体现在工作频段上。由于 IEEE 802.11a 工作在不同于 IEEE 802.11b 的 5.2GHz 频段，避开了当前微波、蓝牙以及大量工业设备广泛采用的 2.4GHz 频段，因此其产品在无线数据传输过程中所受到的干扰大为降低，抗干扰性较 IEEE 802.11b 更为出色。

高达 54Mb/s 的数据传输带宽，是 IEEE 802.11a 的真正意义所在。当 IEEE 802.11b

以其 11Mb/s 的数据传输率满足了一般上网冲浪、数据交换、共享外设等需求的同时，IEEE 802.11a 已经为今后无线宽带网的进一步要求做好了准备，从长远的发展角度来看，其竞争力是不言而喻的。此外，IEEE 802.11a 的无线网络产品较 IEEE 802.11b 有着更低的功耗，这对笔记本电脑以及 PDA 等移动设备来说也有着重大意义。

然而，IEEE 802.11a 的普及也并非一帆风顺的，就像许多新生事物被人们所接受时要面临一些问题一样，IEEE 802.11a 也有其自身的“难言之隐”。

首先，IEEE 802.11a 所面临的难题是来自厂商方面的压力。眼下，IEEE 802.11b 已走向成熟，许多拥有 IEEE 802.11b 产品的厂商会对 IEEE 802.11a 持谨慎态度。二者是竞争还是共存，各厂商的态度莫衷一是。从目前的情况来看，由于这两种技术标准互不兼容，不少厂商为了均衡市场需求，直接将其产品做成了 a+b 的形式，这种做法固然解决了“兼容”问题，但也带来了成本增加的负面因素。其次，相关法律法规的限制，使得 5.2GHz 频段无法在全球各个国家中获得批准和认可。5.2GHz 的高频虽然令 IEEE 802.11a 具有了低干扰的使用环境，但也带来了不利的一面——太空中数以千计的人造卫星与地面站通信也恰恰使用 5.2GHz 频段。此外，欧盟也只允许将 5.2GHz 频率用于其自己制定的另一个无线标准 HiperLAN。

(3) IEEE 802.11g。不可否认，IEEE 802.11g 的诞生为无线网络市场注入了一剂“强心针”，但随之带来的还有无休止的争论，争论的焦点自然是围绕在 IEEE 802.11a 与 IEEE 802.11g 之间。与 IEEE 802.11a 相同的是，IEEE 802.11g 也使用了正交分频多任务(Orthogonal Frequency Division Multiplexing，OFDM)的模块设计，这是其 54Mb/s 高速传输的秘诀。然而不同的是，IEEE 802.11g 的工作频段并不是 IEEE 802.11a 的 5.2GHz，而是坚守在和 IEEE 802.11b 一致的 2.4GHz 频段，这样一来，原先 IEEE 802.11b 使用者所担心的兼容性问题得到了很好的解决，IEEE 802.11g 提供了一个平滑过渡的选择。

既然 IEEE 802.11b 有了 IEEE 802.11a 来替代，无线宽带局域网可谓已经后继有人了，那 IEEE 802.11g 的推出是否多余了呢？答案自然是否定的。除了具备高传输率以及兼容性上的优势外，IEEE 802.11g 所工作的 2.4GHz 频段的信号衰减程度不像 IEEE 802.11a 的5.2GHz 那么严重，并且 IEEE 802.11g 还具备更优秀的“穿透”能力，能适应更加复杂的使用环境。但是先天性的不足(2.4GHz 工作频段)，使得 IEEE 802.11g 和它的前辈 IEEE 802.11b 一样极易受到微波、无线电话等设备的干扰。此外，IEEE 802.11g 的信号比 IEEE 802.11b 的信号能够覆盖的范围要小得多，用户可能需要添置更多的无线接入点才能满足原有使用面积的信号覆盖，这是高速的代价。

(4) IEEE 802.11n。新兴的 802.11n 标准具有高达 600Mb/s 的速率，是下一代无线网络技术，可提供对带宽最为敏感的应用所需要的速率、范围和可靠性。

12.3 无线网络的安全

12.3.1 无线网络与有线网络的区别

由于无线网络通过无线电波在空中传输数据，在数据发射机覆盖区域内几乎所有的无线网络用户都能接触到这些数据。只要具有相同接收频率就可能获取所传递的信息。要将

无线网络环境中传递的数据仅仅传送给一个目标接收者是不可能的。另一方面，由于无线移动设备在存储能力、计算能力和电源供电时间方面的局限性，使得原来在有线环境下的许多安全方案和安全技术不能直接应用于无线环境，例如，防火墙对通过无线电波进行的网络通信起不了作用，任何人在区域范围之内都可以截获和插入数据。计算量大的加密解密算法不适宜用于移动设备等。因此，需要研究新的适合于无线网络环境的安全理论、安全方法和安全技术。与有线网络相比，无线网络所面临的安全威胁更加严重。所有常规有线网络中存在的安全威胁和隐患都依然存在于无线网络中；外部人员可以通过无线网络绕过防火墙，对专用网络进行非授权访问；无线网络传输的信息容易被窃取、篡改和插入；无线网络容易受到拒绝服务攻击（DoS）和干扰；内部员工可以设置无线网卡以端对端模式与外部员工直接连接。此外，无线网络的安全技术相对比较新，安全产品还比较少。以无线局域网（WLAN）为例，移动节点、AP 等每一个实体都有可能是攻击对象或攻击者。由于无线网络在移动设备和传输媒介方面的特殊性，使得一些攻击更容易实施，对无线网络安全技术的研究比有线网络的限制更多，难度更大。无线网络在信息安全方面有着与有线网络不同的特点，具体表现在以下 4 个方面。

1. 无线网络的开放性使得其更容易受到恶意攻击

无线链路使得网络更容易受到从被动窃听到主动干扰的各种攻击。有线网络的网络连接是相对固定的，具有确定的边界，攻击者必须物理接入网络或经过几道防线，如防火墙和网关，才能进入有线网络。这样通过对接入端口的管理可以有效地控制非法用户的接入。而无线网络则没有一个明确的防御边界，攻击者可能来自四面八方和任意节点，每个节点必须面对攻击者直接或间接的攻击。无线网络的这种开放性带来了非法信息截取、未授权信息服务等一系列的信息安全问题。

2. 无线网络的移动性使得安全管理难度更大

有线网络的用户终端与接入设备之间通过线缆连接着，终端不能在大范围内移动，对用户的管理比较容易。而无线网络终端不仅可以在较大范围内移动，而且还可以跨区域漫游，这意味着移动节点没有足够的物理防护，从而易被窃听、破坏和劫持。攻击者可能在任何位置通过移动设备实施攻击，而在全球范围内跟踪一个特定的移动节点是很难做到的；另外，通过网络内部已经被入侵的节点实施攻击而造成的破坏更大，更难检测到。因此，对无线网络移动终端的管理要困难得多，无线网络的移动性带来了新的安全管理问题，移动节点及其体系结构的安全性更加脆弱。

3. 无线网络动态变化的拓扑结构使得安全方案的实施难度更大

有线网络具有固定的拓扑结构，安全技术和方案容易实现。而在无线网络环境中，动态的、变化的拓扑结构。缺乏集中管理机制，使得安全技术更加复杂。另外，无线网络环境中做出的许多决策是分散的，而许多网络算法必须依赖所有节点的共同参与和协作。缺乏集中管理机制意味着攻击者可能利用这一弱点实施新的攻击来破坏协作算法。

4. 无线网络传输信号的不稳定性带来无线通信网络的鲁棒性问题

有线网络的传输环境是确定的，信号质量稳定，而无线网络随着用户的移动其信道特性是变化的，会受到干扰、衰落、多径、多普勒频谱等多方面的影响，造成信号质量波动较大，甚至无法通信。因此，无线网络传输信道的不稳定性带来了无线通信网络的鲁棒性问题。此外，移动计算引入了新的计算和通信行为，这些行为在固定或有线网络中很少出现。例如，移动用户通信能力不足，其原因是链路速度慢、带宽有限、成本较高、电池能量有限等。而无连接操作和依靠地址运行的情况只出现在移动无线环境中。因此，有线网络中的安全措施不能对付基于这些新的应用而产生的攻击。无线网络的脆弱性是由于其媒体的开放性、终端的移动性、动态变化的网络拓扑结构、协作算法、缺乏集中监视和管理点以及没有明确的防线造成的。因此，在无线网络环境中，在设计实现一个完善的无线网络系统时，除了考虑在无线传输信道上提供完善的移动环境下的多业务服务平台外，还必须考虑其安全方案的设计，这包括用户接入控制设计、用户身份认证方案设计、用户证书管理系统的设计、密钥协商及密钥管理方案的设计等。

12.3.2 无线网络面临的安全问题

由于无线局域网采用公共的电磁波作为载体，电磁波能够穿过天花板、玻璃、楼层、砖、墙等物体，因此在一个无线局域网接入点（Access Point）所服务的区域中，任何一个无线客户端都可以接收到此接入点的电磁波信号，这样就可能包括一些恶意用户也能接收到其他无线数据信号。这样恶意用户在无线局域网中相对于在有线局域网当中，去窃听或干扰信息就容易得多。

无线网络所面临的安全威胁主要有以下 6 类。

1. 网络窃听

一般说来，大多数网络通信都是以明文（非加密）格式出现的，这就会使处于无线信号覆盖范围之内的攻击者可以乘机监视并破解（读取）通信。这类攻击是企业管理员面临的最大的安全问题。如果没有基于加密的强有力的安全服务，数据就很容易在空气中传输时被他人读取并利用。

2. 中间人欺骗

在没有足够的安全防范措施的情况下，是很容易受到利用非法 AP 进行的中间人欺骗攻击的。解决这种攻击通常的做法是采用双向认证方法（即网络认证用户，同时用户也认证网络）和基于应用层的加密认证（如 HTTPS＋Web）。

3. WEP 破解

现在互联网上存在一些程序，能够捕捉位于 AP 信号覆盖区域内的数据包，收集到足够的 WEP 弱密钥加密的包，并进行分析以恢复 WEP 密钥。根据监听无线通信的机器速度、WLAN 内发射信号的无线主机数量，以及由于 802.11 帧冲突引起的 IV 重发数量，最快可以在两个小时内攻破 WEP 密钥。

4. MAC 地址欺骗

即使 AP 启用了 MAC 地址过滤，使未授权的黑客的无线网卡不能连接 AP，这并不意味着能阻止黑客进行无线信号侦听。通过某些软件分析截获的数据，能够获得 AP 允许通信的 STA MAC 地址，这样黑客就能利用 MAC 地址伪装等手段入侵网络了。

5. 地址欺骗和会话拦截

由于 802.11 无线局域网对数据帧不进行认证操作，攻击者可以通过欺骗帧去重定向数据流和使 ARP 表变得混乱，通过非常简单的方法，攻击者可以轻易获得网络中站点的 MAC 地址，这些地址可以在恶意攻击时使用。

除通过欺骗帧进行攻击外，攻击者还可以通过截获会话帧发现 AP 中存在的认证缺陷，通过监测 AP 发出的广播帧发现 AP 的存在。然而，由于 802.11 没有要求 AP 必须证明自己真是一个 AP，攻击者很容易装扮成 AP 进入网络，通过这样的 AP，攻击者可以进一步获取认证身份信息从而进入网络。在没有采用 802.11i 对每一个 802.11 MAC 帧进行认证的技术前，通过会话拦截实现的网络入侵是无法避免的。

6. 高级入侵

一旦攻击者进入无线网络，就将成为进一步入侵其他系统的起点。很多网络都有一套经过精心设置的安全设备作为网络的外壳，以防止非法攻击，但是在外壳保护的网络内部却是非常脆弱和容易受到攻击的。无线网络通过简单配置就可快速地接入网络主干，但这样会使网络暴露在攻击者面前。即使有一定边界安全设备的网络，同样也会使网络暴露出来从而遭到攻击。

12.3.3 常用的无线网络安全技术

1. 服务集标识符

通过对多个无线接入点(Access Point，AP)设置不同的 SSID，并要求无线工作站出示正确的 SSID 才能访问 AP，这样就可以允许不同群组的用户接入，并对资源访问的权限进行区别限制。因此可以认为 SSID 是一个简单的口令，从而提供一定的安全，但如果配置 AP 向外广播其 SSID，那么安全程度还将下降。由于一般情况下，用户自己配置客户端系统，所以很多人都知道该 SSID，很容易共享给非法用户。目前有的厂家支持“任何(any)”SSID 方式，只要无线工作站在任何 AP 范围内，客户端都会自动连接到 AP，这将跳过 SSID 安全功能。

2. 物理地址过滤

由于每个无线工作站的网卡都有唯一的物理地址，因此可以在 AP 中手工维护一组允许访问的 MAC 地址列表，实现物理地址过滤。这个方案要求 AP 中的 MAC 地址列表必须随时更新，可扩展性差；而且 MAC 地址在理论上可以伪造，因此这也是较低级别的授权认证。物理地址过滤属于硬件认证，而不是用户认证。这种方式要求 AP 中的 MAC 地址列

表必须随时更新，目前都是手工操作；如果用户增加，则扩展能力很差，因此只适合于小型网络规模。

3. 连线对等保密

在链路层采用 RC4 对称加密技术，用户的加密密钥必须与 AP 的密钥相同时才能获准存取网络的资源，从而防止非授权用户的监听以及非法用户的访问。WEP 提供了 40 位(有时也称为 64 位)和 128 位长度的密钥机制，但是它仍然存在许多缺陷，例如一个服务区内的所有用户都共享同一个密钥，一个用户丢失钥匙将使整个网络不安全。而且 40 位的钥匙在今天很容易被破解；钥匙是静态的，要手工维护，扩展能力差。目前为了提高安全性，建议采用 128 位加密钥匙。

4. Wi-Fi 保护接入

WPA(Wi-Fi Protected Access)是继承了 WEP 基本原理而又解决了 WEP 缺点的一种新技术。由于加强了生成加密密钥的算法，因此即便收集到分组信息并对其进行解析，也几乎无法计算出通用密钥。其原理为根据通用密钥，配合表示电脑 MAC 地址和分组信息顺序的编号，分别为每个分组信息生成不同的密钥。然后与 WEP 一样将此密钥用于 RC4 加密处理。通过这种处理，所有客户端的所有分组信息所交换的数据将由各不相同的密钥加密而成。无论收集到多少这样的数据，要想破解出原始的通用密钥几乎是不可能的。WPA 还追加了防止数据中途被篡改的功能和认证功能。由于具备这些功能，WEP 中此前备受指责的缺点得以全部解决。WPA 不仅是一种比 WEP 更为强大的加密方法，而且有更为丰富的内涵。作为 802.11i 标准的子集，WPA 包含了认证、加密和数据完整性校验三个组成部分，是一个完整的安全性方案。

5. 国家标准

WAPI(WLAN Authentication and Privacy Infrastructure)，即无线局域网鉴别与保密基础结构，它是针对 IEEE 802.11 中的 WEP 协议安全问题，在中国无线局域网国家标准 GB 15629.11 中提出的 WLAN 安全解决方案。同时本方案已由 ISO/IEC 授权的机构 IEEE Registration Authority 审查并获得认可。它的主要特点是采用基于公钥密码体系的证书机制，真正实现了移动终端(MT)与无线接入点(AP)间的双向鉴别。用户只要安装一张证书就可在覆盖 WLAN 的不同地区漫游，方便用户使用。与现有计费技术兼容的服务，可实现按时计费、按流量计费、包月等多种计费方式。AP 设置好证书后，无须再对后台的 AAA 服务器进行设置，安装、组网便捷，易于扩展，可满足家庭、企业、运营商等多种应用模式。

6. 端口访问控制技术

该技术也是用于无线局域网的一种增强型网络安全解决方案。当无线工作站 STA 与无线访问点 AP 关联后，是否可以使用 AP 的服务要取决于 802.1x 的认证结果。如果认证通过，则 AP 为 STA 打开这个逻辑端口，否则不允许用户上网。802.1x 要求无线工作站安装 802.1x 客户端软件，无线访问点要内嵌 802.1x 认证代理，同时它还作为 Radius 客户端，

将用户的认证信息转发给 Radius 服务器。802.1x 除提供端口访问控制能力之外，还提供基于用户的认证系统及计费，特别适合于公共无线接入解决方案。

对于个人用户来讲，具体的防范措施包括：

(1) 建立用户认证，路由器或中继器设备不要使用厂商默认的用户名和密码，应将其修改，而且要定期进行变更。

(2) 数据加密，开启无线网络加密设置，即使在无线网络上传输的数据被截取了，黑客也没办法(或者是说没那么容易)解读数据。此外要注意的是，如果家庭网络中同时存在多个无线网络设备，这些设备的加密技术应该选取同一个。如果需要全面的安全保障，多层加密更好。

(3) 长时间不用的情况下关闭网络。为了不给黑客可乘之机，在长时间不使用网络的情况下，应该手动关闭网络功能；或者在路由器内设置在没有数据传输量的情况下断开连接，在路由器允许的设备接入的情况下，自动连接网络。此外，如果用一台无线路由器，而是用有线连接，那么大可不必开启无线功能；为了保证家庭网络和信息的安全，对无线设备的信息和工作原理进行全面了解，也是防范黑客攻击的一种途径。

(4) 开启网络内每一台设备的网络防火墙。一旦网络被攻破，电脑的防火墙将会起到积极的抵御作用，给信息提供一层安全防护。

(5) 设置 MAC 地址过滤。基本上每一个网络接点设备都有一个独一无二的 MAC 地址，所有路由器/中继器等路由设备都会跟踪所有经过它们的数据包源 MAC 地址。通常，许多这类设备都提供对 MAC 地址的操作，这样可以建立自己的标准通过 MAC 地址列表，防止非法设备(主机等)接入网络。

(6) 为网络设备分配静态 IP，在成员很固定的家庭网络中，可以通过为网络成员设备分配固定的 IP 地址，然后再在路由器上设定允许接入设备 IP 地址列表，从而可以有效地防止非法入侵。

(7) 确定位置，隐藏好家庭路由器或中继器。若出现相邻网络覆盖，影响网络传输，一旦发生这种情况，就需要为路由器或中继器设置一个不同于邻居网络的频段(Channel)。根据自己的家庭，选择好合适的有效范围的路由器或中继器，并选择好其安放的位置，一般来讲，安置在家庭最中间的位置是最合适的。

(8) 启用有线等效加密(Wired Equivalent Privacy，WEP)。WEP 是 IEEE 802.11b 标准中最基本的用于无线网络安全的协议。它不仅可以对无线网络访问者的身份进行识别，防止未授权用户的访问，而且可以对网络传输的数据随机生成密钥进行加密，保护数据的安全。目前 WEP 已被发现有明显漏洞，是一个容易被攻击的协议，攻击者可以通过一些专业攻击工具将其轻易破密，但是对非专业人士而言要想攻破它也绝非易事，何况对于早期的无线网络设备只能使用 WEP 加密。

(9) 启用 Wi-Fi 保护访问(Wi-Fi Protected Access，WPA)。由于 WEP 的弱点而研究产生了 WPA，它包括 WPA 和 WPA2 两个标准。WPA 采用一种基于 TKIP 的方法对密钥进行加密，并包括了可扩展身份验证协议(EAP)以保证只有授权客户端才能访问；WPA2 采用了更高级的 AES 加密方法和动态改变的密钥，让密钥更难攻克。

(10) 设置独特的服务集标识符(Service Set Identifier，SSID)。通常每个无线网络都有一个 SSID，它是无线接入的身份标识。无线客户端只有知道这个 SSID 才能进入该网络。

如果接入点使用的是默认的 SSID,那么任何客户端都可以与该接入点连接;如果接入点使用的 SSID 太简单或普通,也容易让攻击者推测出而进行攻击。因此对一个无线网络应该采取的最基本的措施就是设置一个独特的 SSID。

(11) 禁用 SSID 广播。如果无线网络开启了 SSID 广播,那么在其无线信号的有效范围内,任何无线客户端都能利用无线网络扫描工具接收到该 SSID 号,并使用该网络。如此一来,该网络将面临极大的安全威胁。因此,应采取措施禁用 SSID 广播,它不仅不会影响合法用户的正常使用,而且在其他人的可用网络列表中它将是不可见的。

(12) 根据 MAC 地址对客户端进行过滤。

MAC 地址是指介质访问控制(Media Access Control,MAC)地址,是厂商生产的网卡的地址,通俗点说就是网卡的物理地址,每一台设备的 MAC 是唯一的。因此如果通过 MAC 地址过滤,只允许指定的网卡访问,禁止未经授权的用户入侵无线网络,将会使网络多一层安全防卫。

(13) 基于端口的访问控制和认证协议 802.1X。802.1X 使用标准安全协议(例如 RADIUS)提供集中的用户标识、身份验证、动态密钥管理。802.1X 对连接到交换机端口上的客户端进行认证,只有通过验证的客户端,才能使用该端口,并利用密钥进行通信,否则拒绝连接请求,这将大大减小无线网络的安全风险。

(14) 防火墙(Firewall)技术。防火墙是指隔离在本地网络与外界网络之间的一道执行控制策略的防御系统。它对网络之间传输的数据包依照一定的安全策略进行检查,以决定通信是否被允许,对外屏蔽内部网络的信息、结构和运行状况,并提供单一的安全和审计的安装控制点,从而达到保护内部网络的信息不被外部非授权用户访问和过滤不良信息的目的。目前防火墙技术主要包括包过滤技术、应用代理技术和状态检测技术,可以实现对输入进行筛选、防止内部信息的外泄、限制内部用户活动、对网络使用情况进行记录、监控等。但是它仍然存在缺陷,例如它不能防范不经过它的攻击;不能防范来自内部网络的攻击;只能识别与特征数据匹配的信息,如果攻击者使用恶意代码或攻击伪装,只要能成功避开特征匹配,就能成功通过防火墙等。

(15) 入侵检测系统(Intrusion Detection System,IDS)。入侵检测,即对入侵行为的检测,目前也已经用于无线网络。它通过对网络流量进行收集与分析,并与 IDS 检测器中的攻击特征文件进行对比,从而寻找出违反安全策略的入侵行为并进行报警。由此可见 IDS 中的特征文件是尤为重要的,因此必须保证特征文件的灵活性并及时更新。IDS 仍有其不足,例如它只能检测攻击,而不能阻止攻击。它不能在入侵行为发生之前预报警。它的特征文件的管理和维护较难。

(16) 虚拟专用网(Virtual Private Network,VPN)技术。虚拟专用网(VPN)指的是在一个公共网络上构建的一个临时的、私有的、安全的连接,是一种通过逻辑划分而非物理划分的方式,从公共网络中隔离出来的网络,主要采用的技术包括隧道技术、加密技术、密钥管理技术、访问控制技术及使用者与设备身份认证技术。目前虚拟专用网技术也用于无线环境中,主要是针对安全要求高的大型无线网络。但是它面对的最大威胁是一旦 VPN 内的人员发起攻击,人们将无能为力。即便如此,使用了 VPN 的无线网络仍是更安全的,也是更难被攻破的。

(17) 不允许自动连接功能。许多设备会自动连接无线网络,并且是在不通知用户的情

况下连接,这也是一个安全隐患,应该将其禁止。

习题

1. 无线网络的调制方式包括什么?
2. 常用的无线网络安全技术包括什么?
3. 无线网络与有线网络的区别是什么?
4. 无线网络都面临哪些安全问题?

参考文献

[1] 斯坦普.信息安全原理与实践(第 2 版).北京:清华大学出版社,2013.

[2] 乌斯利.信息安全完全参考手册(第 2 版).北京:清华大学出版社,2014.

[3] 冯登国,赵险峰.信息安全技术概论(第 2 版).北京:电子工业出版社,2014.

[4] 斯廷森(Stinson D R).密码学原理与实践(第三版).冯登国,等,译.北京:电子工业出版社,2009.

[5] 《三思·科学》第二期,2001,8.

[6] 福罗赞(Forouzan B A).密码学与网络安全.马振晗,贾军保,译.北京:清华大学出版社,2009.

[7] 胡向东,魏琴芳.应用密码学教程.北京:电子工业出版社,2005.

[8] 张焕国,王张宜.密码学引论(第二版).武汉:武汉大学出版社,2009.

[9] 卢开澄.计算机密码学——计算机网络中的数据保密与安全(第 3 版).北京:清华大学出版社,2003.

[10] 福罗赞.密码学与网络安全(中文导读英文版).北京:清华大学出版社,2009.

[11] 谷利泽,郑世慧,杨义先.现代密码学教程.北京:北京邮电大学出版社,2009.

[12] 卡哈特.密码学与网络安全(第 2 版).北京:清华大学出版社,2009.

[13] 何大可.现代密码学.北京:人民邮电出版社,2009.

[14] 杨波.现代密码学.北京:清华大学出版社,2003.

[15] 赖溪松,肖国镇.计算机密码学及其应用.北京:国防工业出版社,2000.

[16] 段钢.加密与解密(第三版).北京:电子工业出版社,2008.

[17] 吴宗成.密码学与信息安全.台北:国立台湾科技大学出版社,2006.

[18] 刘玉珍,王丽娜,傅建明.密码编码学与网络安全:原理与实践(第三版).北京:电子工业出版社,2004.

[19] Richard Spillman.经典密码学与现代密码学.叶阮健,译.北京:清华大学出版社,2005.

[20] Massey J L. Shift-Register Synthesis and BCH Decoding. IEEE Tran,on Information theory,1969,15(1):92-97.

[21] 张焕国,刘玉珍.密码学引论.武汉:武汉大学出版社,2003.

[22] 冯登国,裴定一.密码学导引.北京:科学出版社,1999.

[23] 王衍波,薛通.应用密码学.北京:机械工业出版社,2003.

[24] 宋震.密码学.北京:中国水利水电出版社,2002.

[25] William Stallings.密码编码学与网络安全:原理与实践(第二版).杨明,胥光辉,等,译.北京:电子工业出版社,2001.

[26] Wenbo Mao.现代密码学理论与实践.王继林,伍前红,等,译.北京:电子工业出版社,2004.

[27] 张仕斌,何大可.PKI 安全认证体系的研究.计算机应用研究,2005,22(7):97-130.

[28] 孙淑玲.应用密码学.北京:清华大学出版社,2004 .

[29] 汤惟.密码学与网络安全技术基础.北京:机械工业出版社,2004.

[30] 无线安全:看无线网络各类加密模式.赛迪网,2010-9-21. http://www. v6online. com/html/Network/Security/2010/921/3425. html.

[31] 中国互联网站发展状况及其安全报告.赛迪网,2015. http://news. ccidnet. com/art/1032/20150323/5796229_2. html.

[32] 2015 信息安全趋势和任务.中国计算机报,2015. http://www. vsharing. com/k/net/2015-1/709688. html.